GEORGIA INSTITUTE OF TECHNOLOGY
OFFICE OF CONTRACT ADMINISTRATION
SPONSORED PROJECT INITIATION

Date: 6/29/79

Project Title: Control Systems for Interfacing Solar and Bio-Mass Fuels in Agricultural Dryers

Project No: E-21-662

Project Director: Dr. J. H. Schlag

Sponsor: U. S. Department of Agriculture; Science and Education Administration Agricultural Research - Southern Region; New Orleans, LA 70153

Agreement Period: From 6/13/79 Until 6/30/80

Type Agreement: Specific Cooperative Agreement No. 58-7B30-9-83

Amount:

$35,861	USDA
34,006	GIT (E-21-344)
$69,867	TOTAL

Reports Required: Quarterly Progress Reports; Final Technical Report

Sponsor Contact Person (s):

Technical Matters

Authorized Departmental Officer's Designated Representative (ADODR)
Dr. James L. Butler, SEA Sponsoring Scientist
United States Department of Agriculture
Science and Education Administration
Agricultural Research
Georgia Coastal Plain Experiment Station
Tifton, Georgia 31794

Contractual Matters
(thru OCA)

U.S. Department of Agriculture
Science and Education Administration
Agricultural Research -Southern Region
701 Loyola Avenue
P. O. Box 53326
New Orleans, LA 70153

Dr. E. L. Kendrick
Authorized Departmental Officer

Ernest R. George
Research Contracts & Grants Officer

NOTE: Follow-on to E-21-665

Defense Priority Rating: none

Assigned to: Electrical Engineering (School/Laboratory)

COPIES TO:

Project Director
Division Chief (EES)
School/Laboratory Director
Dean/Director--EES
Accounting Office
Procurement Office
Security Coordinator (OCA)
Reports Coordinator (OCA)

Library, Technical Reports Section
EES Information Office
EES Reports & Procedures
Project File (OCA)
Project Code (GTRI)
Other Dr. A. P. Sheppard/VPR

CA-3 (3/76)

GEORGIA INSTITUTE OF TECHNOLOGY
OFFICE OF CONTRACT ADMINISTRATION

SPONSORED PROJECT TERMINATION

Date: October 27, 1980

Project Title: Control Systems for Interfacing Solar and Bio-Mass Fuels in Agricultural Dryers

Project No: E-21-662

Project Director: Dr. J. H. Schlag

Sponsor: U. S. Department of Agriculture; Science and Education Administration Agricultural Research - Southern Region; New Orleans, LA 70153

Effective Termination Date: 10/16/80 (Final Report Submitted)

Clearance of Accounting Charges: - - - - - -

Grant/Contract Closeout Actions Remaining:

X Final Invoice ~~and Closing Documents~~

_ Final Fiscal Report

X Final Report of Inventions

_ Govt. Property Inventory & Related Certificate (No Property Clause)

_ Classified Material Certificate

_ Other ______

NOTE: Follow-on project is E-21-631/Schlag/EE

Assigned to: Electrical Engineering (School/Laboratory)

COPIES TO:

Project Director
Division Chief (EES)
School/Laboratory Director
Dean/Director–EES
Accounting Office
Procurement Office
Security Coordinator (OCA)
Reports Coordinator (OCA)

Library, Technical Reports Section
EES Information Office
Project File (OCA)
Project Code (GTRI)
Other Mr. C. E. Smith

CA–4 (1/79)

QUARTERLY REPORT

Submitted to the
Department of Energy
Through the U.S. Department of Agriculture

from

Georgia Institute of Technology
225 North Avenue, N.W.
Atlanta, Georgia 30332

CONTROL SYSTEMS FOR INTERFACING SOLAR AND BIOMASS FUELS IN AGRICULTURAL DRYERS

Principal Investigator

Dr. J. H. Schlag
Professor
Electrical Engineering
(404) 894-2934
SSN 239-524-477

Project Manager

Dr. A. P. Sheppard
Associate Vice President
for Research
(404) 894-4826
SSN 254-524-258

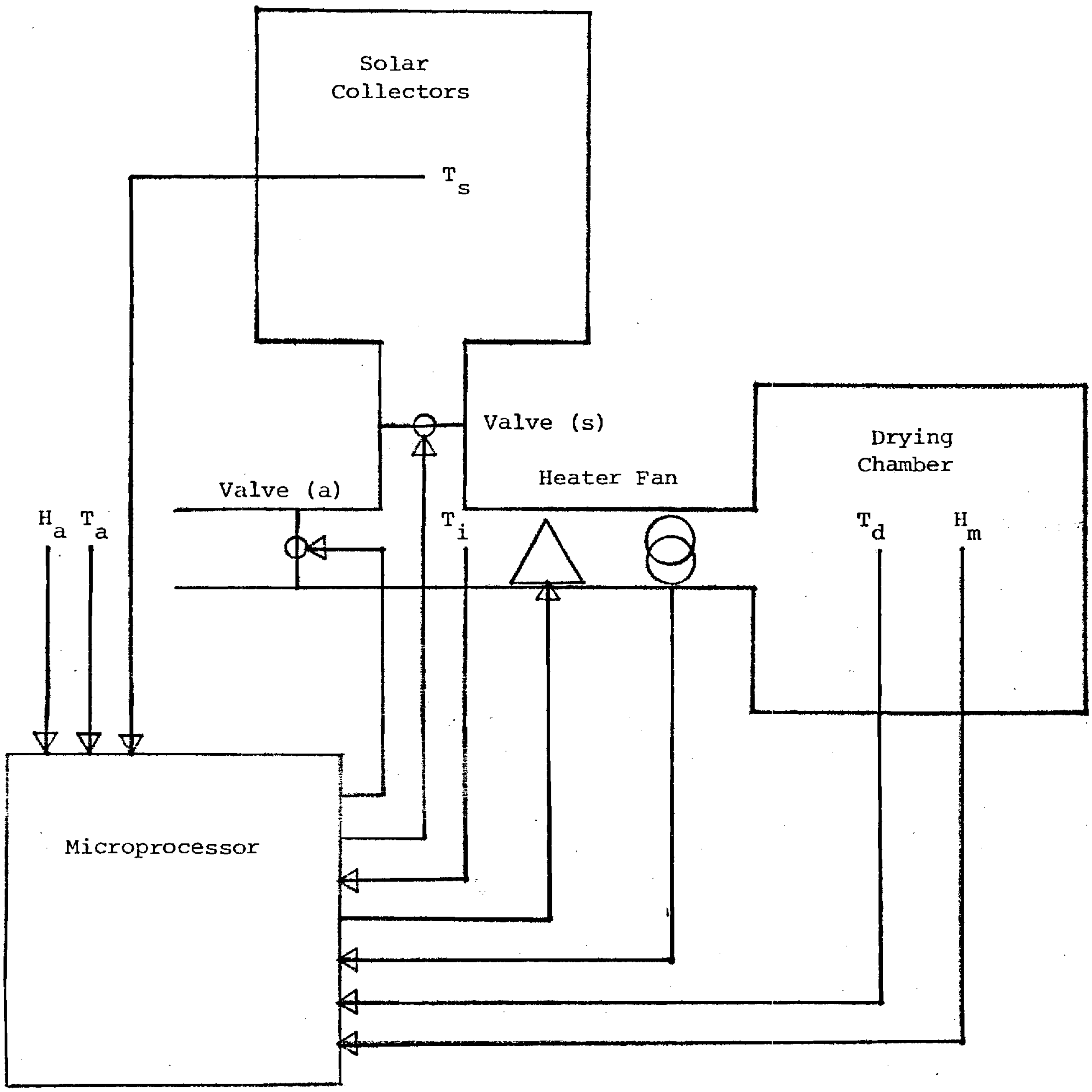
October 1979

INTRODUCTION

Previous work has shown that solar energy is a feasible alternative to fossil fuels for crop drying. However, the greatest drawback to solar energy is that it is not continuously available, and other energy sources must be used as backup. Research is being conducted into the use of biomass to generate methane and alcohol which could be used as backup fuels for solar dryers. It is the objective of this paper to examine the interfacing problem between a solar dryer and a biomass-fueled backup system.

The equipment setup is shown in Figure 1. The valves to ambient air and the solar collectors will be controlled by a microprocessor, as will the amount of biomass gas inlet. The inputs to the microprocessor will include ambient humidity, ambient air temperature, solar collector temperature, intermediate air temperature, the state of the blower fan (on or off), the drying temperature of the air, and the humidity of the drying chamber; the outputs will be the controls on the valve openings and gas inlet. The model which has been developed is essentially a static description of the relationships involved. However, because of the large time-span involved in drying crops (of the order of 24 hours) and the slow response time of valve mechanisms, static relationships can be employed by the microprocessor in an iterative manner, effectively producing a slow-dynamic model.

The first question addressed by the model is how the ideal drying temperature for a particular substance determines the length of time required to bring the moisture content from some initial level to a pre-



Solar
Collectors
Ts
Valve (s)
Valve (a)
Heater Fan
Drying
Chamber
Ha Ta
Ti
Td
Hm
Microprocessor

FIGURE 1

determined final level.

The second question addressed is what combination of ambient air, solar-heated air, and additional biomass thermal power is most appropriate and economical under prevailing climatic conditions, i.e., ambient temperature, insolation, etc.

Finally the desired rates of flow of ambient air and solar-heated air must be related to the angular openings of the two valves and to any resistance to air flow (drag) inherent in the two sources.

DEFINITION OF SYMBOLS

Let symbols be defined as follows:

Q_{req} = heat energy required for drying

Q_{sup} = heat energy supplied by ambient air, solar collectors, biomass fuel, or some combination of these three

m = mass of matter to be dried

M = mass of air used in drying

μ_i = initial water fraction in matter to be dried

μ_f = final water fraction in matter to be dried

c_w = specific heat capacity of water = 1.0 Kcal/KgC°

c_a = specific heat capacity of ambient air

h_v = heat of vaporization of water

h_c = heat of combustion of biomass fuel (alcohol or methane)

T_a = temperature (°C) of ambient air

T_s = temperature of solar collector system

T_d = ideal drying temperature for a particular type of matter

T_m = initial temperature of matter to be dried

T_i = temperature of warm air before heating by biomass fuel

H_a = ambient humidity

ρ_a = density of ambient air

H_m = humidity of drying chamber

$\dot{V}_d$ = total volume flow rate provided by fan, assumed constant

$\dot{V}_a$ = volume flow rate of ambient air

$\dot{V}_s$ = volume flow rate of solar-heated air

Δt = time duration of drying process

$\dot{q}_{req}$ = thermal power required from biomass fuel

$\dot{q}_{sup}$ = thermal power supplied by biomass fuel

$\dot{v}$ = volume flow rate for gas (alcohol or methane) in the biomass heater

θ = angle of opening of valve to ambient air

ϕ = angle of opening of valve to solar collector system

r = radius of duct pipe

EXPLANATION OF MODEL

I. From a consideration of the expressions for Q_{req} and Q_{sup} the relation between T_d and Δt can be shown:

$$Q_{req} = (\mu_i - \mu_f)m\, c_w(T_d - T_m) + (\mu_i - \mu_f)mh_v(H_m, T_d) \tag{1}$$

Note that vaporization occurs at $T_d < 100°C$, and that the heat of vaporization will be a function of both humidity and temperature.

$$\begin{aligned} Q_{sup} &= M\, c_a(H_a)T_d \\ &= \rho_a\, \dot{V}_d\, \Delta t\, c_a(H_a)T_d \end{aligned} \tag{2}$$

Here note that the specific heat of air will be a function of humidity.

By recognizing that Q_{sup} must equal Q_{req}, and that ρ_a, $\dot{V}_d$, and $c_a(H_a)$ are not adjustable, it follows that T_d determines Δt.

II. The discussion now concerns how usage of ambient air, solar-heated air and biomass fuel is determined.

A. If $T_a = T_d$, then open only the valve to ambient air, leaving valve to solar system closed and biomass heater off. In this case $\dot{V}_a = \dot{V}_d$, and $\dot{V}_s = 0$.

B. If $T_a < T_d$ and $T_s = T_d$, then open only the valve to the solar system, leaving valve to ambient air closed and biomass heater off. In this case $\dot{V}_s = \dot{V}_d$, and $\dot{V}_a = 0$.

C. If $T_a < T_d$ and $T_s > T_d$, then leave biomass heater off and open both valves (to ambient air and solar system) to appropriate degrees so that

$$\begin{cases} \dot{V}_a + \dot{V}_s = \dot{V}_d \\ T_a\dot{V}_a + T_s\dot{V}_s = T_d\dot{V}_d \end{cases} \tag{3}$$

whose solution is

$$\dot{V}_a = \dot{V}_d\left(\frac{T_s - T_d}{T_s - T_a}\right)$$

$$\dot{V}_s = \dot{V}_d\left(\frac{T_d - T_a}{T_s - T_a}\right) \tag{4}$$

D. Finally, if $T_a < T_d$ and $T_s < T_d$, then extra heat energy will have to be added by the biomass-powered heater.

In this case

$$\dot{V}_a + \dot{V}_s = \dot{V}_d$$

$$T_a\dot{V}_a + T_s\dot{V}_s = T_i\dot{V}_d < T_d\dot{V}_d \tag{5}$$

$$\text{and } T_i < T_d$$

In order to raise T_i to T_d, $\dot{q}_{req}$ Kcal/sec must be added, where

$$\dot{q}_{req} = \frac{d}{dt}\left[M\ c_a(H_a)(T_d - T_i)\right]$$

$$= c_a(H_a)(T_d - T_i)\ \rho_a\ \dot{V}_d \tag{6}$$

The thermal power supplied is given by

$$\dot{q}_{sup} = h_c \dot{v} \tag{7}$$

Since $\dot{q}_{sup}$ must equal $\dot{q}_{req}$, T_i determines $\dot{v}$.

E. Although there will be a thermocouple positioned to measure the temperature T_i of the air mixed from the ambient air valve and the solar system valve, it is a simple matter to calculate it from T_a, $\dot{V}_a$, T_s, and $\dot{V}_s$.

$$T_i = \frac{T_a \dot{V}_a + T_s \dot{V}_s}{\dot{V}_a + \dot{V}_s} \tag{8}$$

Even though it is intuitively apparent in this form as a weighted average, the expression can also be found by using a simple heat transfer calculation.

III. It remains to discuss how the volume flow rate for a valve depends on angular opening and on any resistive structures within. When the two valves are used together, these flow rates will be coupled, as stated earlier.

$$\dot{V}_a + \dot{V}_s = \dot{V}_d$$

It will be useful to consider an analogous electrical circuit. Let the resistance within the ambient air system be R_a, that within the solar system be R_s. Let the resistance offered by the ambient air valve (open to angle θ) be R_θ, that offered by the solar system valve (open to angle ϕ) be R_ϕ. Let the total air flow rate $\dot{V}_d$ be represented by the total current in the circuit i_d, and the flow rates in the "parallel" valves by i_a and i_s. The source of emf (pressure) is E. A diagram of the circuit is given in Figure 2. The solutions to this circuit for $i_a(\dot{V}_a)$ and $i_s(\dot{V}_s)$ are given by:

$$i_a = i_d \left(\frac{R_s + R_\phi}{R_a + R_\theta + R_s + R_\phi}\right) \tag{9}$$

$$i_s = i_d \left(\frac{R_a + R_\theta}{R_a + R_\theta + R_s + R_\phi}\right)$$

It is clear from the solutions to this analogous electrical circuit that the drag factors in the two air paths of the drying system will be closely coupled. Since there is no <u>a priori</u> way of modeling the air drag quantities represented by R_a, R_θ, R_s, and R_ϕ, empirical measurements will be used to obtain their relative magnitudes. One useful relationship can be obtained, however. It seems reasonable that the amount of drag presented to the air flow by a valve will be proportional to the projected area of the valve on the plane of the opening; thus

$$R_\theta \ \alpha \ \pi r^2 \cos \theta \text{ and } R_\phi \ \alpha \ \pi r^2 \cos \phi \tag{10}$$

Where it is assumed that the radius of the duct pipe r will be the same throughout the system. Nevertheless, an exact understanding of the magnitudes of these four drag factors will not be necessary because of the iterative control procedure employed: an initial setting of the air valves and gas inlet will be determined by the input parameters; the intermediate temperature and ultimate drying temperature will be checked; the air valves and gas inlet will be re-set, if necessary; then temperatures will be re-checked and the iteration process continued.

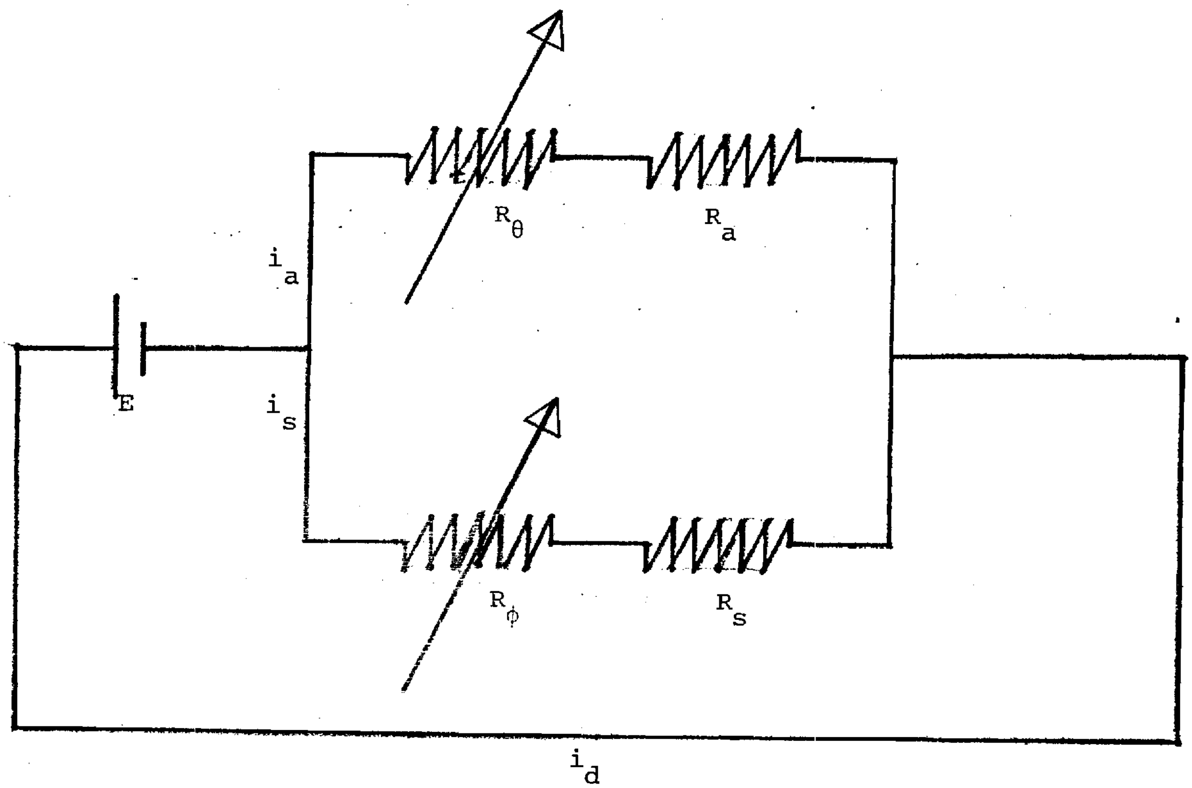


FIGURE 2

CONCLUSION

The discussion has treated the component parts of this interfacing problem by simple modeling procedures. Although the mathematical relationships could be further explored in various places, it is felt that the iterative nature of the control process will be adequate to refine the approximations made by the model.

QUARTERLY REPORT

Submitted to the
Department of Energy
Through the U.S. Department of Agriculture

from

Georgia Institute of Technology
225 North Avenue, N.W.
Atlanta, Georgia 30332

CONTROL SYSTEMS FOR INTERFACING SOLAR AND BIOMASS FUELS IN AGRICULTURAL DRYERS

Principal Investigator	Project Manager
Dr. J. H. Schlag	Dr. A. P. Sheppard
Professor	Associate Vice President for Research
Electrical Engineering	
(404) 894-2934	(404) 894-4826
SSN 239-524-477	SSN 254-524-258

January 1980

INTRODUCTION

The previous quarterly report included a description of a theoretical model for mixing solar heated air, ambient air, and heat from an auxiliary source such as a biomass convertor. Functional equations were provided to describe the physics involved in the operation of the system. The model represents a means for establishing an ideal environment for drying crops, as well as optimizing the use of solar and thermal power under variable climatic conditions.

EXPERIMENTAL MODEL

In order to test the theory put forth in the report of last October, an experimental model has been designed and is under construction. A functional diagram of the equipment involved is presented in Figure 1, which uses the same symbology as the earlier report. Most of the symbol definitions are given again in the next section.

The ductwork for the model is made of galvanized aluminum "stove pipe." Thermocouple temperature probes and air flow meters are positioned as shown. The air from the chamber representing the solar collectors will be heated by an alternate source so that the entire unit can be assembled and operated within the laboratory. This is necessary since the lab contains the facilities which are essential for developing the microprocessor software.

This software will contain the algorithm to calculate the desired setting for the dampers based on the ambient temperature, the temperature of the air from the solar collector (representing climatic conditions), and the selected range for the dryer temperature. The equations presented in the previous report are incorporated in the algorithm.

The valves shown in Figure 1 are actually dampers that are 90 degrees out-of-phase; that is, when one is completely open permitting maximum air flow, the other is closed for minimum air flow through the pipe. The two dampers are physically linked so that as one damper is closing the other is opening. The stepping motor, which provides the power to operate the dampers, receives signals from the microprocessor indicating whether the motor should rotate clockwise or counterclockwise and by how many increments.

SOFTWARE

A flowchart of the microprocessor's control program is shown in Appendix A. The symbols are defined as:

H_M - initial humidity of the material to be dried

T_S - temperature of the "solar" heated air

H_A - ambient humidity

T_A - temperature of the ambient air

T_D - temperature of the drying chamber

θ - angle of the damper in the solar duct (0° is minimum air flow)

ℓ - lower limit of dryer temperature range

u - upper limit of dryer temperature range

a - program variable for iteratively calculating θ

α - intermediate angle in calculating proper θ.

When the program is entered initially, the temperature is measured. If the temperature of the solar heated air is within the desired

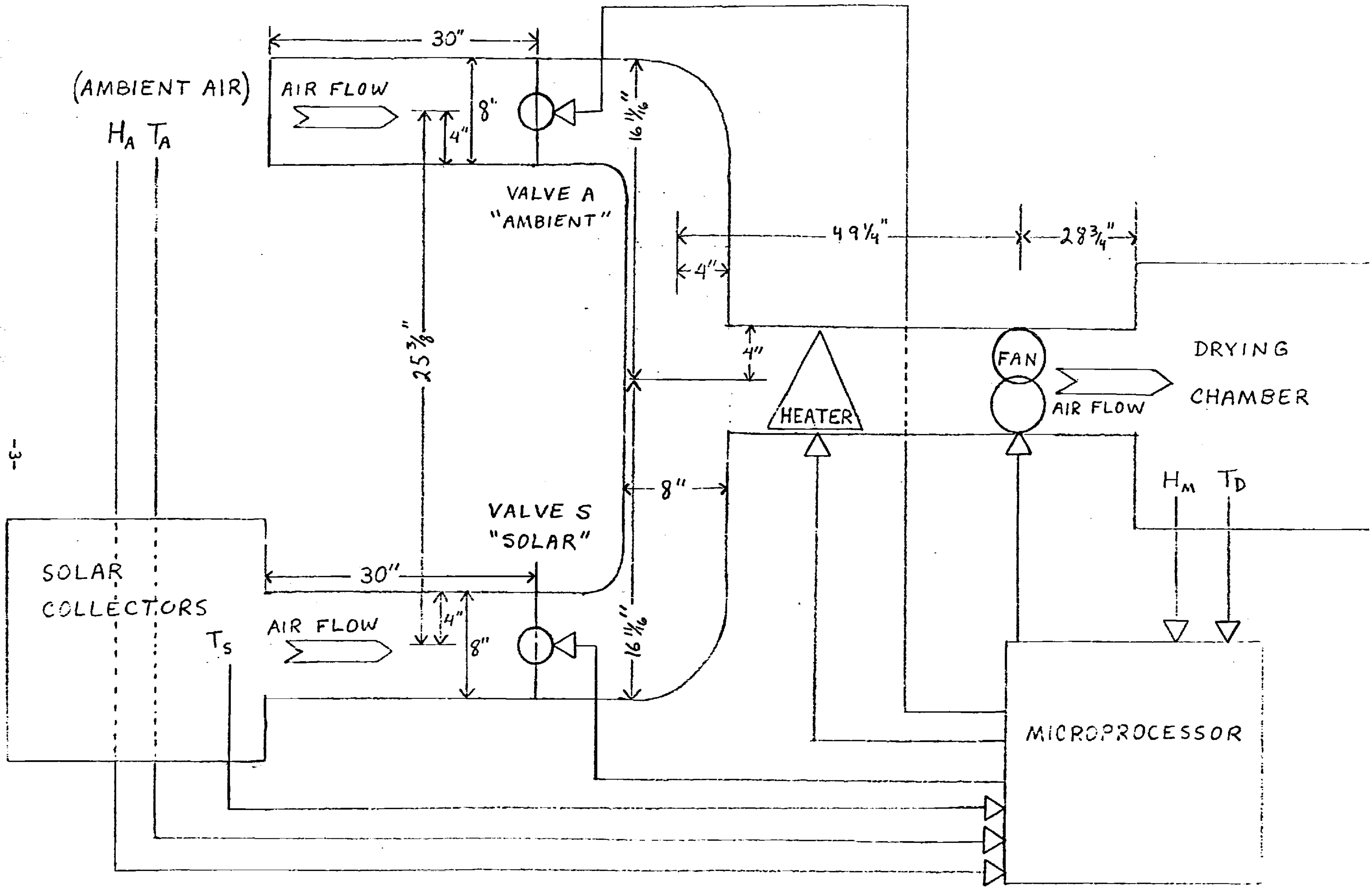


Figure 1. Experimental Model (not to scale)

range for the drying chamber, then the proper dryer temperature T_D can theoretically be attained by using solar heated air alone without intermixing any ambient air. Program operation will remain in this loop (labeled "Max Solar" in the flowchart) until T_S deviates outside the desired range.

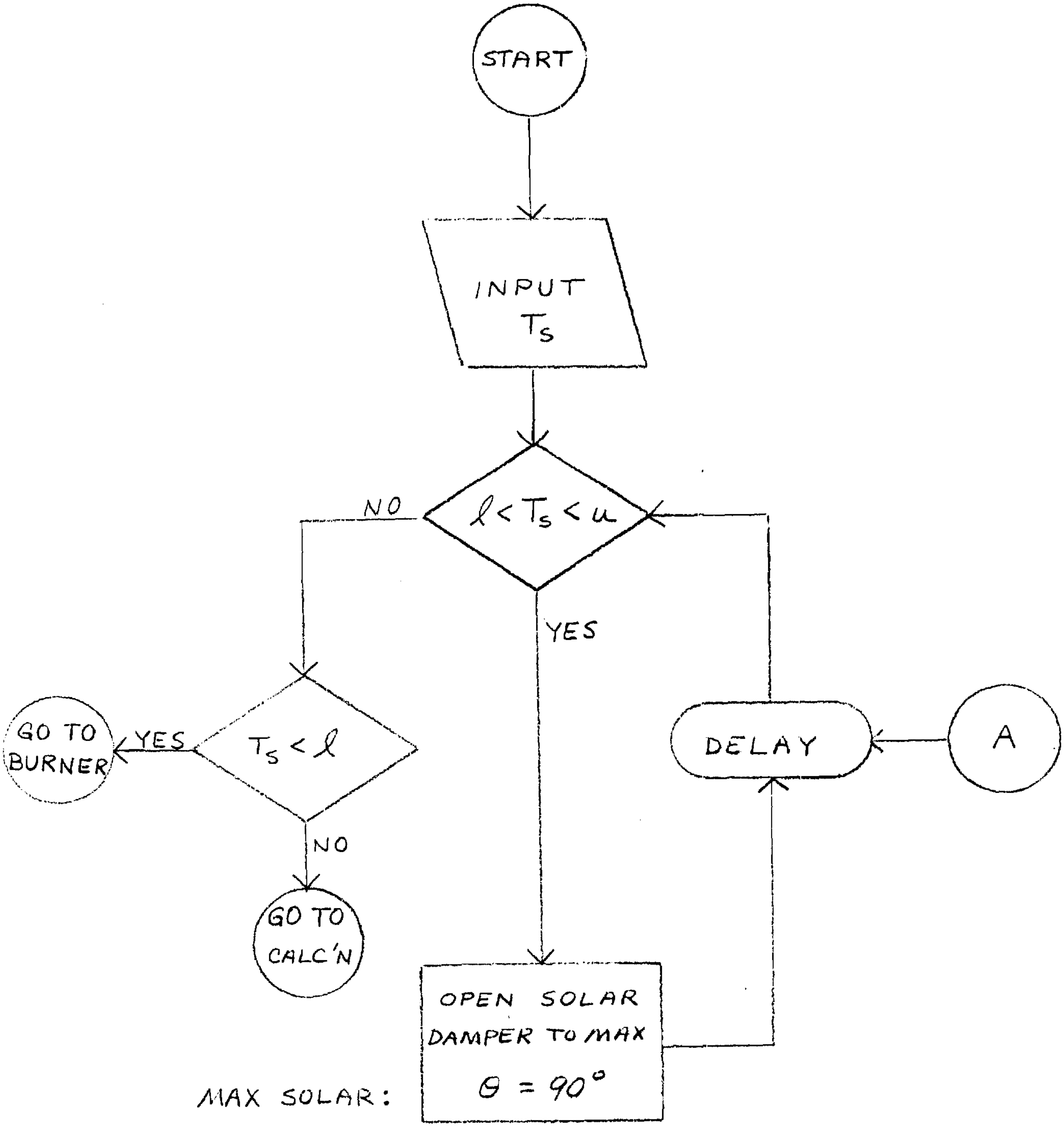
If T_S is found to be below the lower limit, the program will go to the "burner" loop. It is assumed that the ambient air will always be colder than the solar air. If T_S is too low, then the burner must be turned on and left on until the dryer temperature is well within the selected range. Both T_S and T_D are checked within the execution of this loop. If T_S should increase, the program will exit the loop; otherwise, if T_D falls below the lower limit the burner will be turned on again.

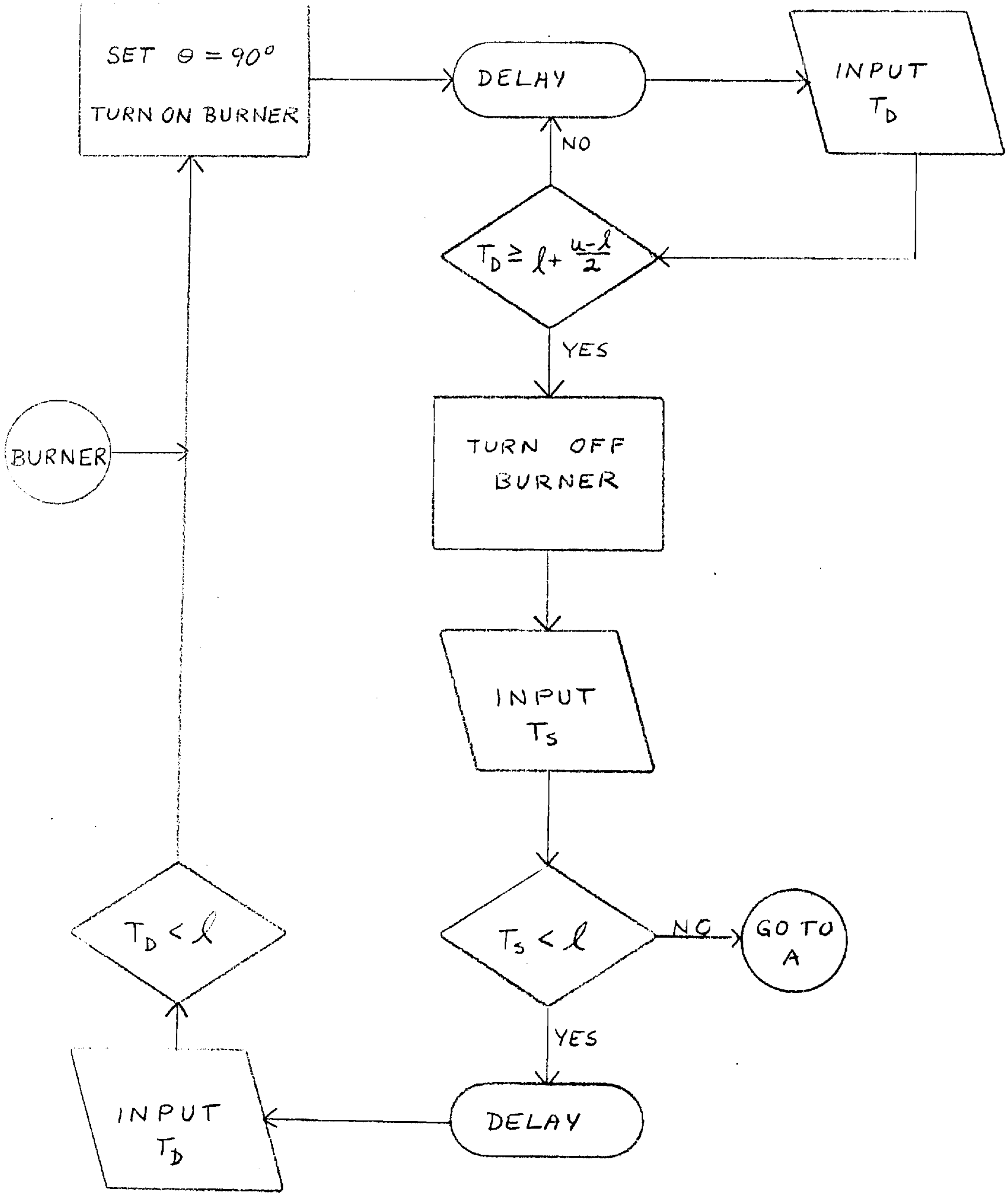
For the case where T_S is above the upper limit, the program will enter the "calculation" loop, which determines the proper value for theta based on the measured temperatures of T_A and T_S. This procedure uses the equations of the October report to select the angular setting for the dampers. According to the theory, after the dampers have been adjusted the dryer temperature should stabilize within the acceptable range. However, if the dryer temperature does not reach an acceptable level after a certain time lapse, the microprocessor will signal the stepping motor to alter theta accordingly. After each incremental adjustment and a time delay, the dryer temperature will be measured again. This routine will continue until T_D falls within the desired limits, after which the temperature T_D is monitored for any further variations.

A trial program has been written in PL/M and assembled on an Intel MDS-II development system. Appendix B contains an entire program listing with the assembly code for each PL/M statement. During system testing, this program will be stored in the read-only memory of an SBC (single board computer) model 8020. It can be changed at any time to accommodate modifications to the algorithm, control signals, or program loops.

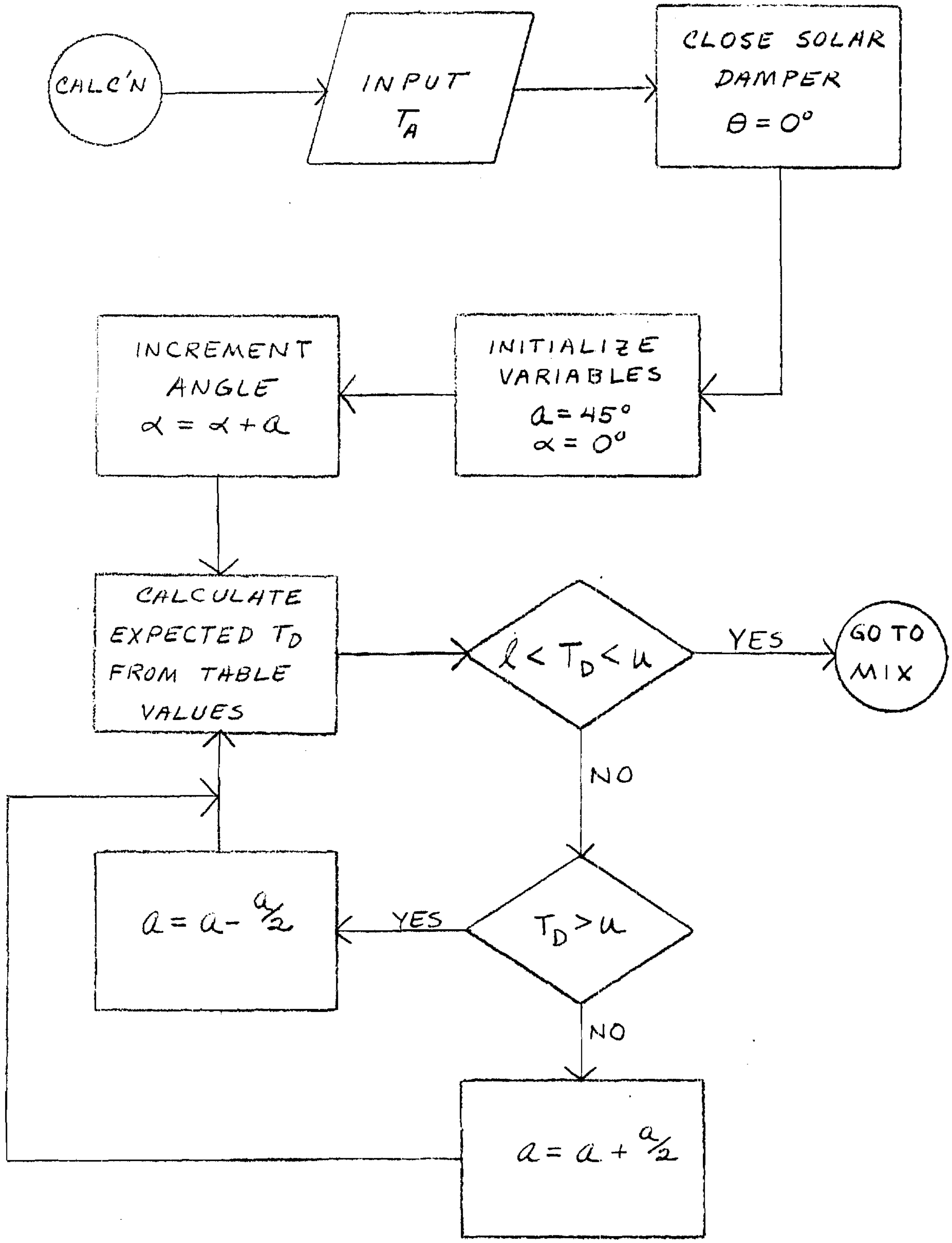
APPENDIX A

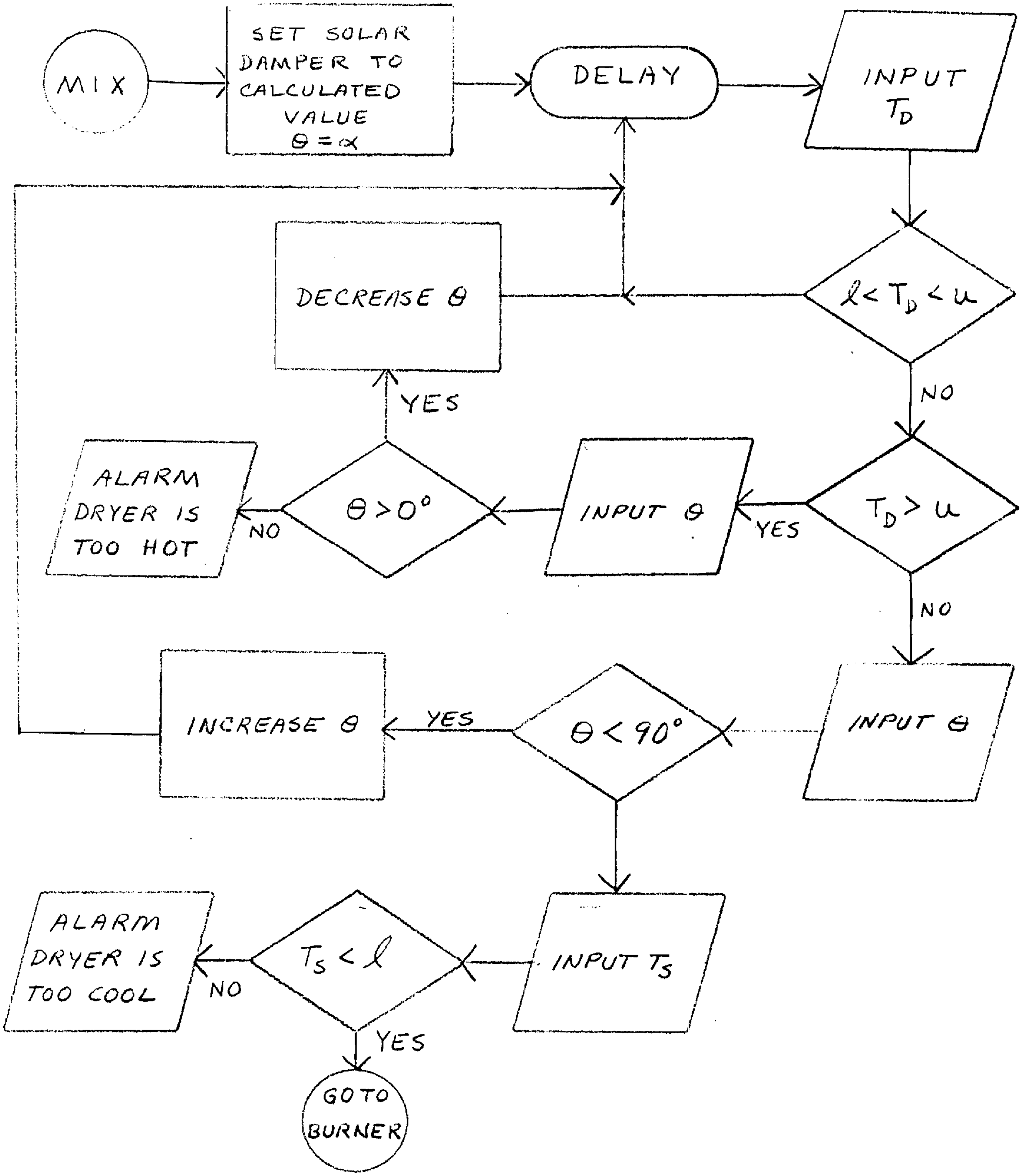
Flowchart for Controlling Dampers and Burners


START
INPUT T_s
$l < T_s < u$
NO
YES
GO TO BURNER
YES
$T_s < l$
NO
GO TO CALC'N
DELAY
A
OPEN SOLAR DAMPER TO MAX
$\theta = 90°$
MAX SOLAR:



SET $\theta = 90°$
TURN ON BURNER
DELAY
INPUT T_D
NO
$T_D \geq \ell + \frac{u-\ell}{2}$
YES
TURN OFF BURNER
BURNER
INPUT T_S
$T_D < \ell$
$T_S < \ell$
NO
GO TO A
YES
INPUT T_D
DELAY


CALC'N
INPUT T_A
CLOSE SOLAR DAMPER $\theta = 0°$
INITIALIZE VARIABLES $a = 45°$ $\alpha = 0°$
INCREMENT ANGLE $\alpha = \alpha + a$
CALCULATE EXPECTED T_D FROM TABLE VALUES
$l < T_D < u$
YES
GO TO MIX
NO
$T_D > u$
YES
$a = a - a/2$
NO
$a = a + a/2$


MIX
SET SOLAR DAMPER TO CALCULATED VALUE θ = α
DELAY
INPUT T_D
$l < T_D < u$
NO
$T_D > u$
YES
INPUT θ
θ > 0°
YES
DECREASE θ
NO
ALARM DRYER IS TOO HOT
NO
INPUT θ
θ < 90°
YES
INCREASE θ
INPUT T_S
$T_S < l$
NO
ALARM DRYER IS TOO COOL
YES
GOTO BURNER

APPENDIX B

Program Listing for Process Control

```
ISIS-II PL/M-80 V3.0 COMPILATION OF MODULE USDA
OBJECT MODULE PLACED IN :F1:USDA.OBJ
COMPILER INVOKED BY:  PLM80 :F1:USDA.SRC CODE



   1          USDA:DO;
   2    1       DECLARE (TS,TA,TD) ADDRESS;
   3    1       DECLARE (ALT,L,J,I,U,A,THETA,N) BYTE INITIAL(1,70,0,0,80,0,0
   4    1       DECLARE KS(50) ADDRESS DATA(3282,1689,1157,891,731,624,547,4
                  408,378,353,331,313,296,282,269,258,247,238,229,221,
                  213,206,200,194,188,183,178,173,168,163,159,155,151,147,
                  143,140,136,132,129,126,122,119,116,113,109,106,103,100);
   5    1       DECLARE KA(50) ADDRESS DATA(100,103,106,109,113,116,119,122,1
                  132,136,140,143,147,151,155,159,163,168,173,178,183,188,1
                  200,206,213,221,229,238,247,258,269,282,296,313,331,353,
                  408,444,489,547,634,731,892,1157,1689,3282);

   6    1       FWD:PROCEDURE;
                                                     ; STATEMENT # 6
                            ; PROC  FWD
   7    2             OUTPUT(0)=1;
                                                     ; STATEMENT # 7
         0354  3E01              MVI      A,1H
         0356  D300              OUT      0H
   8    2             CALL TIME(100);       /*10MS DELAY*/
                                                     ; STATEMENT # 8
         0358  3E64              MVI      A,64H
         035A  CD0000            CALL     @P0105
   9    2             OUTPUT(0)=0;
                                                     ; STATEMENT # 9
         035D  3E00              MVI      A,0H
         035F  D300              OUT      0H
  10    2             CALL TIME(100);       /*PULSES @ 50HZ*/
                                                     ; STATEMENT # 10
         0361  3E64              MVI      A,64H
         0363  CD0000            CALL     @P0105
  11    2           END FWD;
                                                     ; STATEMENT # 11
         0366  C9                RET

  12    1       DELAY:PROCEDURE;
                                                     ; STATEMENT # 12
                            ; PROC  DELAY
  13    2             DO I=1 TO 4;
                                                     ; STATEMENT # 13
         0367  210900            LXI      H,I
         036A  3601              MVI      M,1H
                        @21:
         036C  3E04              MVI      A,4H
         036E  210900            LXI      H,I
         0371  BE                CMP      M
         0372  DA9603            JC       @22
  14    3             DO J=1 TO 250;        /*10 SEC DELAY*/
                                                     ; STATEMENT # 14
         0375  210800            LXI      H,J
         0378  3601              MVI      M,1H
```

```
                              @23:
          037A   3EFA                 MVI      A,0FAH
          037C   210800               LXI      H,J
          037F   BE                   CMP      M
          0380   DA8F03               JC       @24
 15    4                   CALL TIME(100);
                                                            ; STATEMENT # 15
          0383   3E64                 MVI      A,64H
          0385   CD0000               CALL     @P0105
 16    4                   END;
                                                            ; STATEMENT # 16
          0388   210800               LXI      H,J
          038B   34                   INR      M
          038C   C27A03               JNZ      @23
                              @24:
 17    3                   END;
                                                            ; STATEMENT # 17
          038F   210900               LXI      H,I
          0392   34                   INR      M
          0393   C26C03               JNZ      @21
                              @22:
 18    2               END DELAY;
                                                            ; STATEMENT # 18
          0396   C9                   RET

 19    1               DEG90:PROCEDURE;
                                                            ; STATEMENT # 19
                              ; PROC  DEG90
 20    2                   DO WHILE(SHR((NOT INPUT(0)),1));
                                                            ; STATEMENT # 20
                              @25:
          0397   DB00                 IN       0H
          0399   2F                   CMA
          039A   B7                   ORA      A
          039B   1F                   RAR
          039C   1F                   RAR
          039D   D2A603               JNC      @26
 21    3                       CALL FWD;
                                                            ; STATEMENT # 21
          03A0   CD5403               CALL     FWD
 22    3                   END;
                                                            ; STATEMENT # 22
          03A3   C39703               JMP      @25
                              @26:
 23    2               END DEG90;
                                                            ; STATEMENT # 23
          03A6   C9                   RET

 24    1               BKWD:PROCEDURE;
                                                            ; STATEMENT # 24
                              ; PROC  BKWD
 25    2                   OUTPUT(0)=2;     /*RAISE CONTROL LINE FOR BKWD PULSES*/
                                                            ; STATEMENT # 25
          03A7   3E02                 MVI      A,2H
          03A9   D300                 OUT      0H
 26    2                   CALL TIME(100);  /*10 MS DELAY*/
                                                            ; STATEMENT # 26
```

```
          03AB  3E64              MVI     A,64H
          03AD  CD0000            CALL    @P0105
 27    2             OUTPUT(0)=0;     /*LOWER CONTROL LINE*/
                                                     ; STATEMENT # 27
          03B0  3E00              MVI     A,0H
          03B2  D300              OUT     0H
 28    2             CALL TIME(100);
                                                     ; STATEMENT # 28
          03B4  3E64              MVI     A,64H
          03B6  CD0000            CALL    @P0105
 29    2          END BKWD;
                                                     ; STATEMENT # 29
          03B9  C9                RET


 30    1          CO:PROCEDURE(CHAR) EXTERNAL;
 31    2             DECLARE CHAR BYTE;
 32    2          END CO;

 33    1          START:DO WHILE (NOT INPUT(0)); /*BIT 0 INDICATES 0 DEGREES*/
                                                     ; STATEMENT # 33
          00C8  310000            LXI     SP,@STACK$ORIGIN
                           START:
                           @27:
          00CB  DB00              IN      0H
          00CD  1F                RAR
          00CE  DAD700            JC      @28
 34    2               CALL BKWD;
                                                     ; STATEMENT # 34
          00D1  CDA703            CALL    BKWD
 35    2               END;
                                                     ; STATEMENT # 35
          00D4  C3CB00            JMP     @27
                           @28:

 36    1          SOLAR:  TS=INPUT(1);    /*MEASURE  TEMP OF SOLAR HEATED AIR*
                                                     ; STATEMENT # 36
                           SOLAR:
          00D7  DB01              IN      1H
          00D9  6F                MOV     L,A
          00DA  2600              MVI     H,0
          00DC  220000            SHLD    TS
 37    1             IF(TS>U)THEN GOTO MIX;
                                                     ; STATEMENT # 37
          00DF  3A0A00            LDA     U
          00E2  CD0000            CALL    @P0094
          00E5  D2EB00            JNC     @1
                                                     ; STATEMENT # 38
          00E8  C35201            JMP     MIX
                           @1:
 39    1             CALL DEG90;
                                                     ; STATEMENT # 39
          00EB  CD9703            CALL    DEG90
 40    1             IF(TS<L)THEN GOTO BURNER;
                                                     ; STATEMENT # 40
          00EE  3A0700            LDA     L
          00F1  110000            LXI     D,TS
```

```
        00F4  CD0000              CALL    @P0101
        00F7  D2FD00              JNC     @2
                                                          ; STATEMENT # 41
        00FA  C30301              JMP     BURNER
                            @2:
 42   1                      ELSE CALL DELAY;
                                                          ; STATEMENT # 42
        00FD  CD6703              CALL    DELAY
                            @3:
 43   1                  GOTO SOLAR;
                                                          ; STATEMENT # 43
        0100  C3D700              JMP     SOLAR

 44   1              BURNER: OUTPUT(0)=4;    /*RAISE CONTROL LINE TO LIGHT BURNER*
                                                          ; STATEMENT # 44
                            BURNER:
        0103  3E04                MVI     A,4H
        0105  D300                OUT     0H
 45   1                  DO WHILE(INPUT(3)<L+(U-L)/2);    /*INPUT(3) IS TD*/
                                                          ; STATEMENT # 45
                            @29:
        0107  DB03                IN      3H
        0109  210700              LXI     H,L
        010C  F5                  PUSH    PSW     ; 1
        010D  3A0A00              LDA     U
        0110  96                  SUB     M
        0111  5F                  MOV     E,A
        0112  1600                MVI     D,0
        0114  210200              LXI     H,2H
        0117  CD0000              CALL    @P0029
        011A  2A0700              LHLD    L
        011D  2600                MVI     H,0
        011F  19                  DAD     D
        0120  F1                  POP     PSW     ; 1
        0121  CD0000              CALL    @P0094
        0124  D22D01              JNC     @30
 46   2                      CALL DELAY;
                                                          ; STATEMENT # 46
        0127  CD6703              CALL    DELAY
 47   2                  END;
                                                          ; STATEMENT # 47
        012A  C30701              JMP     @29
                            @30:
 48   1                  OUTPUT(0)=0;    /*TURN OFF BURNER*/
                                                          ; STATEMENT # 48

        012D  3E00                MVI     A,0H
        012F  D300                OUT     0H
 49   1                  LOOP:IF (INPUT(1)<L)THEN CALL DELAY;    /*IF TS<L*/
                                                          ; STATEMENT # 49
                            LOOP:
        0131  DB01                IN      1H
        0133  210700              LXI     H,L
        0136  BE                  CMP     M
        0137  D24001              JNC     @4
                                                          ; STATEMENT # 50
        013A  CD6703              CALL    DELAY
```

```
          013D  C34301          JMP     @5
                         @4:
 51   1                          ELSE GOTO SOLAR;     /*IF TS>L*/
                                                ; STATEMENT # 51
          0140  C3D700          JMP     SOLAR
                         @5:
 52   1                   IF(INPUT(3)<L)THEN GOTO BURNER; /*IF TD<L & TS<L*/
                                                ; STATEMENT # 52
          0143  DB03            IN      3H
          0145  210700          LXI     H,L
          0148  BE              CMP     M
          0149  D24F01          JNC     @6
                                                ; STATEMENT # 53
          014C  C30301          JMP     BURNER
                         @6:
 54   1                          ELSE GOTO LOOP; /*IF TD>L & TS<L*/
                                                ; STATEMENT # 54
          014F  C33101          JMP     LOOP
                         @7:

 55   1          MIX:     TA=INPUT(2);
                                                ; STATEMENT # 55
                         MIX:
          0152  DB02            IN      2H
          0154  6F              MOV     L,A
          0155  2600            MVI     H,0
          0157  220200          SHLD    TA
 56   1                 IF((INPUT(0) AND 1)=1)THEN THETA=0;
                                                ; STATEMENT # 56
          015A  DB00            IN      0H
          015C  E601            ANI     1H
          015E  FE01            CPI     1H
          0160  C26801          JNZ     @8
                                                ; STATEMENT # 57
          0163  210C00          LXI     H,THETA
          0166  3600            MVI     M,0H
                         @8:
 58   1                 IF((INPUT(0) AND 2)=2)THEN THETA=50;
                                                ; STATEMENT # 58
          0168  DB00            IN      0H
          016A  E602            ANI     2H
          016C  FE02            CPI     2H
          016E  C27601          JNZ     @9
                                                ; STATEMENT # 59

          0171  210C00          LXI     H,THETA
          0174  3632            MVI     M,32H
                         @9:
                        /*ITERATIVE PROCEDURE FOR DETERMINING THEORETICAL SETTIN
 60   1                 N,A=25;
                                                ; STATEMENT # 60
          0176  210D00          LXI     H,N
          0179  3619            MVI     M,19H
          017B  210B00          LXI     H,A
          017E  3619            MVI     M,19H
 61   1                 ALT=1;
                                                ; STATEMENT # 61
```

```
          0180  210600          LXI     H,ALT
          0183  3601            MVI     M,1H
 62   1          ITER:    TD=TS*100/KS(N)+TA*100/KA(N);
                                                ; STATEMENT # 62
                           ITER:
          0185  116400          LXI     D,64H
          0188  2A0000          LHLD    TS
          018B  CD0000          CALL    @P0034
          018E  E5              PUSH    H       ; 1
          018F  2A0D00          LHLD    N
          0192  2600            MVI     H,0
          0194  010000          LXI     B,KS
          0197  29              DAD     H
          0198  09              DAD     B
          0199  4E              MOV     C,M
          019A  23              INX     H
          019B  46              MOV     B,M
          019C  D1              POP     D       ; 1
          019D  CD0000          CALL    @P0030
          01A0  D5              PUSH    D       ; 1
          01A1  116400          LXI     D,64H
          01A4  2A0200          LHLD    TA
          01A7  CD0000          CALL    @P0034
          01AA  E5              PUSH    H       ; 2
          01AB  2A0D00          LHLD    N
          01AE  2600            MVI     H,0
          01B0  016400          LXI     B,KA
          01B3  29              DAD     H
          01B4  09              DAD     B
          01B5  4E              MOV     C,M
          01B6  23              INX     H
          01B7  46              MOV     B,M
          01B8  D1              POP     D       ; 2
          01B9  CD0000          CALL    @P0030
          01BC  E1              POP     H       ; 1
          01BD  19              DAD     D
          01BE  220400          SHLD    TD
 63   1               IF((L<TD) AND (TD<U))THEN DO;
                                                ; STATEMENT # 63
          01C1  3A0700          LDA     L
          01C4  CD0000          CALL    @P0094
          01C7  9F              SBB     A
          01C8  F5              PUSH    PSW     ; 1
          01C9  3A0A00          LDA     U
          01CC  110400          LXI     D,TD
          01CF  CD0000          CALL    @P0101
          01D2  9F              SBB     A
          01D3  C1              POP     B       ; 1
          01D4  48              MOV     C,B
          01D5  A1              ANA     C
          01D6  1F              RAR
          01D7  D20202          JNC     @10
 65   2                               DO WHILE (N<THETA);
                                                ; STATEMENT # 65
                          @31:
          01DA  210C00          LXI     H,THETA
          01DD  3A0D00          LDA     N
```

```
         01E0  BE                  CMP     M
         01E1  D2EE01              JNC     @32
 66   3                                  CALL BKWD;
                                                     ; STATEMENT # 66
         01E4  CDA703              CALL    BKWD
 67   3                                  THETA=THETA-1;
                                                     ; STATEMENT # 67
         01E7  210C00              LXI     H,THETA
         01EA  35                  DCR     M
 68   3                                  END;
                                                     ; STATEMENT # 68
         01EB  C3DA01              JMP     @31
                           @32:
 69   2                                  DO WHILE(N>THETA);
                                                     ; STATEMENT # 69
                           @33:
         01EE  3A0C00              LDA     THETA
         01F1  210D00              LXI     H,N
         01F4  BE                  CMP     M
         01F5  D20202              JNC     @34
 70   3                                  CALL FWD;
                                                     ; STATEMENT # 70
         01F8  CD5403              CALL    FWD
 71   3                                  THETA=THETA+1;
                                                     ; STATEMENT # 71
         01FB  210C00              LXI     H,THETA
         01FE  34                  INR     M
 72   3                                  END;
                                                     ; STATEMENT # 72
         01FF  C3EE01              JMP     @33
                           @34:
 73   2                                END;
                                                     ; STATEMENT # 73
                           @10:

 74   1                  IF(TD<L)THEN DO;
                                                     ; STATEMENT # 74
         0202  3A0700              LDA     L
         0205  110400              LXI     D,TD
         0208  CD0000              CALL    @P0101
         020B  D23F02              JNC     @11
 76   2                            A=(A+ALT)/2;
                                                     ; STATEMENT # 76
         020E  3A0600              LDA     ALT
         0211  210B00              LXI     H,A
         0214  86                  ADD     M
         0215  5F                  MOV     E,A
         0216  1600                MVI     D,0
         0218  210200              LXI     H,2H
         021B  CD0000              CALL    @P0029
         021E  210B00              LXI     H,A
         0221  73                  MOV     M,E
 77   2                            IF(N=23)THEN N=24;
                                                     ; STATEMENT # 77
         0222  3A0D00              LDA     N
         0225  FE17                CPI     17H
         0227  C22F02              JNZ     @12
```

```
                                                          ; STATEMENT # 78
            Ø22A  21ØDØØ           LXI      H,N
            Ø22D  3618             MVI      M,18H
                          @12:
  79   2                          N=N+A;
                                                          ; STATEMENT # 79
            Ø22F  3AØBØØ           LDA      A
            Ø232  21ØDØØ           LXI      H,N
            Ø235  86               ADD      M
            Ø236  77               MOV      M,A
  8Ø   2                          ALT=-ALT;
                                                          ; STATEMENT # 8Ø
            Ø237  3AØ6ØØ           LDA      ALT
            Ø23A  2F               CMA
            Ø23B  3C               INR      A
            Ø23C  32Ø6ØØ           STA      ALT
  81   2                          END;
                                                          ; STATEMENT # 81
                          @11:
  82   1                IF(TD>U)THEN DO;
                                                          ; STATEMENT # 82
            Ø23F  3AØAØØ           LDA      U
            Ø242  21Ø4ØØ           LXI      H,TD
            Ø245  CDØØØØ           CALL     @PØ1Ø3
            Ø248  D27EØ2           JNC      @13
  84   2                          A=(ALT+A)/2;
                                                          ; STATEMENT # 84
            Ø24B  3AØBØØ           LDA      A
            Ø24E  21Ø6ØØ           LXI      H,ALT
            Ø251  86               ADD      M
            Ø252  5F               MOV      E,A
            Ø253  16ØØ             MVI      D,Ø
            Ø255  21Ø2ØØ           LXI      H,2H
            Ø258  CDØØØØ           CALL     @PØØ29
            Ø25B  21ØBØØ           LXI      H,A
            Ø25E  73               MOV      M,E
  85   2                          ALT=-ALT;
                                                          ; STATEMENT # 85
            Ø25F  3AØ6ØØ           LDA      ALT
            Ø262  2F               CMA
            Ø263  3C               INR      A
            Ø264  32Ø6ØØ           STA      ALT
  86   2                          IF(N=27)THEN N=26;
                                                          ; STATEMENT # 86
            Ø267  3AØDØØ           LDA      N
            Ø26A  FE1B             CPI      1BH
            Ø26C  C274Ø2           JNZ      @14
                                                          ; STATEMENT # 87
            Ø26F  21ØDØØ           LXI      H,N
            Ø272  361A             MVI      M,1AH
                          @14:
  88   2                          N=N-A;
                                                          ; STATEMENT # 88
            Ø274  21ØBØØ           LXI      H,A
            Ø277  3AØDØØ           LDA      N
            Ø27A  96               SUB      M
            Ø27B  32ØDØØ           STA      N
```

```
 89   2                      END;
                                                      ; STATEMENT # 89
                        @13:
 90   1                IF(N<>THETA)THEN GOTO ITER;
                                                      ; STATEMENT # 90
          027E  210C00            LXI     H,THETA
          0281  3A0D00            LDA     N
          0284  BE                CMP     M
          0285  CA8B02            JZ      @15
                                                      ; STATEMENT # 91
          0288  C38501            JMP     ITER
                        @15:

 92   1            MAINT:  CALL DELAY;  /*10 SEC DELAY*/
                                                      ; STATEMENT # 92
                        MAINT:
          028B  CD6703            CALL    DELAY
 93   1                TD=INPUT(3);    /*MEASURE DRYER TEMP*/
                                                      ; STATEMENT # 93

          028E  DB03              IN      3H
          0290  6F                MOV     L,A
          0291  2600              MVI     H,0
          0293  220400            SHLD    TD
 94   1            DCR:    IF((TD>U) AND (THETA>0)) THEN DO;
                                                      ; STATEMENT # 94
                        DCR:
          0296  3A0A00            LDA     U
          0299  210400            LXI     H,TD
          029C  CD0000            CALL    @P0103
          029F  9F                SBB     A
          02A0  F5                PUSH    PSW     ; 1
          02A1  3E00              MVI     A,0H
          02A3  210C00            LXI     H,THETA
          02A6  96                SUB     M
          02A7  9F                SBB     A
          02A8  C1                POP     B       ; 1
          02A9  48                MOV     C,B
          02AA  A1                ANA     C
          02AB  1F                RAR
          02AC  D2B602            JNC     @16
 96   2                               THETA=THETA-1;
                                                      ; STATEMENT # 96
          02AF  210C00            LXI     H,THETA
          02B2  35                DCR     M
 97   2                               CALL FWD;
                                                      ; STATEMENT # 97
          02B3  CD5403            CALL    FWD
 98   2                               END;
                                                      ; STATEMENT # 98
                        @16:
 99   1            INCR:   IF((TD<L) AND (THETA<50)) THEN DO;
                                                      ; STATEMENT # 99
                        INCR:
          02B6  3A0700            LDA     L
          02B9  110400            LXI     D,TD
          02BC  CD0000            CALL    @P0101
```

```
         Ø2BF   9F                SBB     A
         Ø2CØ   F5                PUSH    PSW       ; 1
         Ø2C1   3AØCØØ            LDA     THETA
         Ø2C4   D632              SUI     32H
         Ø2C6   9F                SBB     A
         Ø2C7   C1                POP     B         ; 1
         Ø2C8   48                MOV     C,B
         Ø2C9   A1                ANA     C
         Ø2CA   1F                RAR
         Ø2CB   D2D5Ø2            JNC     @17
1Ø1   2                                CALL FWD;
                                                    ; STATEMENT # 1Ø1
         Ø2CE   CD54Ø3            CALL    FWD
1Ø2   2                                THETA=THETA+1;
                                                    ; STATEMENT # 1Ø2
         Ø2D1   21ØCØØ            LXI     H,THETA
         Ø2D4   34                INR     M
1Ø3   2                                END;
                                                    ; STATEMENT # 1Ø3
                           @17:

1Ø4   1                 IF(INPUT(1)<=L)THEN GOTO SOLAR;
                                                    ; STATEMENT # 1Ø4
         Ø2D5   DBØ1              IN      1H
         Ø2D7   4F                MOV     C,A
         Ø2D8   3AØ7ØØ            LDA     L
         Ø2DB   B9                CMP     C
         Ø2DC   DAE2Ø2            JC      @18
                                                    ; STATEMENT # 1Ø5
         Ø2DF   C3D7ØØ            JMP     SOLAR
                           @18:
1Ø6   1                 IF((TD>U) AND (THETA=Ø))THEN GO TO ALARM;
                                                    ; STATEMENT # 1Ø6
         Ø2E2   3AØAØØ            LDA     U
         Ø2E5   21Ø4ØØ            LXI     H,TD
         Ø2E8   CDØØØØ            CALL    @PØ1Ø3
         Ø2EB   9F                SBB     A
         Ø2EC   F5                PUSH    PSW       ; 1
         Ø2ED   3AØCØØ            LDA     THETA
         Ø2FØ   D6ØØ              SUI     ØH
         Ø2F2   D6Ø1              SUI     1
         Ø2F4   9F                SBB     A
         Ø2F5   C1                POP     B         ; 1
         Ø2F6   48                MOV     C,B
         Ø2F7   A1                ANA     C
         Ø2F8   1F                RAR
         Ø2F9   D2FFØ2            JNC     @19
                                                    ; STATEMENT # 1Ø7
         Ø2FC   C31FØ3            JMP     ALARM
                           @19:
1Ø8   1                 IF((TD<L) AND (THETA=5Ø))THEN GOTO ALARM;
                                                    ; STATEMENT # 1Ø8
         Ø2FF   3AØ7ØØ            LDA     L
         Ø3Ø2   11Ø4ØØ            LXI     D,TD
         Ø3Ø5   CDØØØØ            CALL    @PØ1Ø1
         Ø3Ø8   9F                SBB     A
         Ø3Ø9   F5                PUSH    PSW       ; 1
```

```
          030A  3A0C00          LDA     THETA
          030D  D632            SUI     32H
          030F  D601            SUI     1
          0311  9F              SBB     A
          0312  C1              POP     B           ; 1
          0313  48              MOV     C,B
          0314  A1              ANA     C
          0315  1F              RAR
          0316  D21C03          JNC     @20
                                                    ; STATEMENT # 109
          0319  C31F03          JMP     ALARM
                          @20:

110   1               GO TO MAINT;
                                                    ; STATEMENT # 110
          031C  C38B02          JMP     MAINT

111   1           ALARM:  DO I=1 TO 10;
                                                    ; STATEMENT # 111
                          ALARM:
          031F  210900          LXI     H,I
          0322  3601            MVI     M,1H
                          @35:
          0324  3E0A            MVI     A,0AH
          0326  210900          LXI     H,I
          0329  BE              CMP     M
          032A  DA5203          JC      @36
112   2               CALL CO('E');
                                                    ; STATEMENT # 112
          032D  0E45            MVI     C,45H
          032F  CD0000          CALL    CO
113   2               CALL CO('R');
                                                    ; STATEMENT # 113
          0332  0E52            MVI     C,52H
          0334  CD0000          CALL    CO
114   2               CALL CO('R');
                                                    ; STATEMENT # 114
          0337  0E52            MVI     C,52H
          0339  CD0000          CALL    CO
115   2               CALL CO('O');
                                                    ; STATEMENT # 115
          033C  0E4F            MVI     C,4FH
          033E  CD0000          CALL    CO
116   2               CALL CO('R');
                                                    ; STATEMENT # 116
          0341  0E52            MVI     C,52H
          0343  CD0000          CALL    CO
117   2               CALL CO(' ');
                                                    ; STATEMENT # 117
          0346  0E20            MVI     C,20H
          0348  CD0000          CALL    CO
118   2               END;
                                                    ; STATEMENT # 118
          034B  210900          LXI     H,I
          034E  34              INR     M
          034F  C22403          JNZ     @35
                          @36:
```

```
 119   1        END USDA;
                                                  ; STATEMENT # 119
          0352  FB              EI
          0353  76              HLT


MODULE INFORMATION:

     CODE AREA SIZE     = 03BAH     954D
     VARIABLE AREA SIZE = 000EH      14D
     MAXIMUM STACK SIZE = 0006H       6D
     127 LINES READ
     0 PROGRAM ERROR(S)

END OF PL/M-80 COMPILATION
```

GEORGIA INSTITUTE OF TECHNOLOGY
SCHOOL OF ELECTRICAL ENGINEERING
ATLANTA, GEORGIA 30332

TELEPHONE: (404) 894-2934

April 18, 1980

Dr. James L. Butler
Coastal Plain Experiment Station
Tifton, GA 31794

Dear Jim:

Enclosed please find the April 1980 quarterly report. I look forward to seeing you on Thursday, April 24.

Sincerely,

Jay H. Schlag, Ph.D.

JHS:lso

Enclosures

QUARTERLY REPORT

Submitted to the
Department of Energy
Through the U.S. Department of Agriculture

from

Georgia Institute of Technology
225 North Avenue, N.W.
Atlanta, Georgia 30332

CONTROL SYSTEMS FOR INTERFACING SOLAR AND BIOMASS FUELS IN AGRICULTURAL DRYERS

Principal Investigator

Dr. J. H. Schlag
Professor
Electrical Engineering
(404)-894-2934
SSN: 239-52-5477

Project Manager

Dr. A. P. Sheppard
Acting Vice President
for Research
(404)-894-4826
SSN: 254-52-4258

April 1980

Part of the past quarter's work on the hot air mixing system has involved investigation of a suitable method for providing heated air for the prototype unit. One alternative includes a helical spiral of copper tubing to encircle the input air path. The copper tubing would carry the biomass fuel of interest (e.g. methane) and would be perforated at intervals to allow a series of gas flames. Thus the input air would be heated as it flowed into the system. Another alternative consists of a large cubical enclosure in which space heaters would be placed and from which the heated air would be drawn. Still another, and possibly the best, air heating method would use standard furnace grating technology. The valve and ignition system would be controlled together and activated by a single voltage. The advantage in using standard furnace equipment for the prototype unit is that the component parts necessary for an actual solar/biomass agricultural dryer would then be easily available, off-the-shelf items, which would presumably be less expensive than a custom-made system.

Another object of last quarter's effort was the microprocessor control for the hot air mixing system. The 8020 microprocessor unit was connected to the pipe-and-valve system and was used to control the stepping motor which opens and closes the valves in the two air channels. The temperature sensors provided inputs to the computer, and the alarm indicator lights were functional. In addition, the microprocessor software has been modified to provide more interaction with the operator. An ADM-3A CRT terminal has been interfaced to the microprocessor so that the program can output messages which indicate what part of the program is currently being executed. Another aid to prototype testing that has been added is the facility for downloading programs from the MDS-230 development system to the 8020 microprocessor unit. This has eliminated much of the time involved in program development and debugging. With this capability, the software can be loaded into RAM, making it possible for the operator to set breakpoints within the program.

A complete listing of the PLM 80 program is included along with explanatory comments.

```
ISIS-II PL/M-80 V3.Ø COMPILATION OF MODULE USDA
OBJECT MODULE PLACED IN :F1:USDA.OBJ
COMPILER INVOKED BY:  PLM8Ø :F1:USDA.SRC



  1           USDA:DO;           /*DATE:11 APRIL 8Ø*/
  2     1         DECLARE OE4 BYTE INITIAL(ØFFH);
  3     1         DECLARE (L,U,TS,TA,TD) ADDRESS INITIAL(7ØØ,8ØØ);
  4     1         DECLARE (ALT,J,I,A,THETA,N) BYTE INITIAL(1,Ø,Ø,Ø,Ø,Ø);
  5     1         DECLARE KS(5Ø) ADDRESS DATA(3282,1689,1157,891,751.624,547,48
                      4Ø8,378,353,331,313,296,282,269,258,247,238,229,221,
                      213,2Ø6,2ØØ,194,188,183,178,173,168,163,159,155,151,147,
                      143,14Ø,136,132,129,126,122,119,116,113,1Ø9,1Ø6,1Ø3,1ØØ);
  6     1         DECLARE KA(5Ø) ADDRESS DATA(1ØØ,1Ø3,1Ø6,1Ø9,113,116,119,122,1
                      132,136,14Ø,143,147,151,155,159.163,168,173,178,183,188,1
                      2ØØ,2Ø6,213,221,229,238,247,258,269,282,296,313,331,353,3
                      4Ø8,444,489,547,634,751,892,1157,1689,3282);

  7     1         CO:PROCEDURE(CHAR) EXTERNAL;
  8     2            DECLARE CHAR BYTE;
  9     2            END CO;

 1Ø     1         NMOUT:PROCEDURE(ASCII) EXTERNAL;
 11     2               DECLARE ASCII BYTE;
 12     2               END NMOUT;

 13     1         FWD:PROCEDURE;  /*PULSES THE MOTOR CONTROL THAT
                          OPENS THE SOLAR DAMPER(1.6DEG/PULSE)*/

 14     2             OE4=OE4 AND ØEFH;    /*SET BIT 4--PORT E4
                                    HAS INVERTED OUTPUTS*/
 15     2             OUTPUT(ØE4H)=OE4;
 16     2             DO I=1 TO 5Ø;
 17     3             CALL TIME(1ØØ);        /*1Ø MS DELAY*/
 18     3             END;
 19     2             OE4=OE4 OR 1ØH; /*ZERO BIT 4*/
 2Ø     2             OUTPUT(ØE4H)=OE4;
 21     2             DO I=1 TO 5Ø;
 22     3             CALL TIME(1ØØ);
 23     3             END;
 24     2         END FWD;

 25     1         DELAY:PROCEDURE;
 26     2             DO I=1 TO 4;
 27     3             DO J=1 TO 25Ø;         /*1Ø SEC DELAY*/
 28     4             CALL TIME(1ØØ);
 29     4             END;
 3Ø     3             END;
 31     2         END DELAY;

              /*THE FOLLOWING PROCEDURE OPENS SOLAR DAMPER TO MAX POSITION(9ØDE

 32     1         DEG9Ø:PROCEDURE;
 33     2             DO WHILE(SHR((NOT INPUT(ØECH)),1));
 34     3                 CALL FWD;
 35     3             END;
```

```
36   2          END DEG90;

37   1          BKWD:PROCEDURE;       /*PULSES MOTOR CONTROL TO
                              CLOSE SOLAR DAMPER(1.8DEG/PULSE)*/

38   2              OE4=OE4 AND 0F7H;     /*SET BIT 3*/
39   2              OUTPUT(0E4H)=OE4;     /*RAISE CONTROL LINE
                                   FOR BWK(CW) PULSES*/
40   2              DO I=1 TO 50;
41   3              CALL TIME(100);       /*10 MS DELAY*/
42   3              END;
43   2              OE4=OE4 OR 8H;        /*ZERO BIT 3*/
44   2              OUTPUT(0E4H)=OE4;     /*LOWER CONTROL LINE*/
45   2              DO I=1 TO 50;
46   3              CALL TIME(100);
47   3              END;
48   2          END BKWD;


49   1          INIT:OE4=0FFH;             /*INITIALIZE CONTROL WORD
                                  FOR PORT E4*/

50   1                   OUTPUT(0EBH)=98H;
51   1                OUTPUT(0E7H)=89H;

52   1          START:DO WHILE (NOT INPUT(0E8H));/*CLOSE DAMPER TO SOLAR AIR*/
53   2                CALL BKWD;
54   2                END;

55   1          SOLAR:OE4=(0F8H AND OE4) OR 5;
56   1              OUTPUT(0E4H)=OE4;     /*SELECT SOLAR TEMP SENSOR*/
57   1              OUTPUT(0EAH)=1;
58   1              OUTPUT(0EAH)=0;       /*PULSE TO INITIATE CONVERSION*/
59   1              CALL CO(0AH);
60   1              CALL CO(0DH);
61   1              CALL CO('S');
62   1              CALL CO('O');
63   1              CALL CO('L');
64   1              CALL CO('A');
65   1              CALL CO('R');
66   1              CALL CO(0DH);
67   1              CALL CO(0AH);
68   1              TS=(0C0H AND INPUT(0EAH))*4+(NOT INPUT(0E8H));

                /*TS=SOLAR TEMP*/

69   1              CALL NMOUT(HIGH(TS));
70   1              CALL NMOUT(LOW(TS));
71   1              CALL CO(' ');
72   1              CALL CO('T');
73   1              CALL CO('S');
74   1              CALL CO(0DH);
75   1              CALL CO(0AH);
76   1              IF(TS>U)THEN GOTO MIX;
78   1              CALL DEG90;      /*TS<U,OPEN SOLAR DAMPER MAX*/

79   1              IF(TS<L)THEN GOTO BURNER;
```

```
 81   1            ELSE CALL DELAY;     /*ELSE L<TS<U;
                                  STAY IN SOLAR ROUTINE*/
 82   1        GOTO SOLAR;

 83   1     BURNER: OE4=(OE4 AND ØD8H) OR 6;
 84   1        OUTPUT(ØE4H)=OE4;          /*SELECT DRYER TEMP SENSOR AND
                            RAISE CONTROL LINE TO LIGHT BURNER*/
 85   1     INTCON: OUTPUT(ØEAH)=1;
 86   1        OUTPUT(ØEAH)=Ø;       /*INITIATE CONVERSION*/
 87   1        CALL CO('B');
 88   1        CALL CO('U');
 89   1        CALL CO('R');
 90   1        CALL CO('N');
 91   1        CALL CO('E');
 92   1        CALL CO('R');
 93   1        CALL CO(ØDH);
 94   1        CALL CO(ØAH);
 95   1        TD=(ØCØH AND INPUT(ØEAH))*4+(NOT INPUT(ØE8H));

           /*TD=DRYER TEMP*/

 96   1        CALL NMOUT(HIGH(TD));
 97   1        CALL NMOUT(LOW(TD));
 98   1        CALL CO(' ');
 99   1        CALL CO('T');
100   1        CALL CO('D');
101   1        CALL CO(ØDH);
102   1        CALL CO(ØAH);

103   1        IF (TD<L+(U-L)/2) THEN CALL DELAY;
105   1        ELSE GO TO BRNOFF;
106   1        GO TO INTCON;
107   1     BRNOFF: OE4=OE4 OR 2ØH;        /*ZERO BIT 5 OF PORT1*/
108   1        OUTPUT(ØE4H)=OE4;     /*TURN OFF BURNER*/


109   1     LOOP:    OE4=(OE4 AND ØF8H) OR 5;
110   1             OUTPUT(ØE4H)=OE4;     /*SELECT TS*/
111   1        OUTPUT(ØEAH)=1;
112   1        OUTPUT(ØEAH)=Ø;
113   1        CALL CO('L');
114   1        CALL CO('O');
115   1        CALL CO('O');
116   1        CALL CO('P');
117   1        CALL CO(ØAH);
118   1        CALL CO(ØDH);
119   1        TS=(ØCØH AND INPUT(ØEAH))*4+(NOT INPUT(ØE8H));
120   1        CALL NMOUT(HIGH(TS));
121   1        CALL NMOUT(LOW(TS));
122   1        CALL CO(' ');
123   1        CALL CO('T');
124   1        CALL CO('S');
125   1        CALL CO(ØDH);
126   1        CALL CO(ØAH);
127   1        IF (TS<L)THEN CALL DELAY;
129   1                ELSE GOTO SOLAR;
130   1        OE4=(OE4 AND ØF8H) OR 6;
```

```
1                OUTPUT(0E4H)=OE4;            /*SELECT TD*/
1                OUTPUT(0EAH)=1;
1                OUTPUT(0EAH)=0;
1                CALL TIME(100);
1                TD=(0C0H AND INPUT(0EAH))*4+(NOT INPUT(0E9H));
1                CALL NMOUT(HIGH(TD));
1                CALL NMOUT(LOW(TD));
1                CALL CO(' ');
1                CALL CO('T');
1                CALL CO('D');
1                CALL CO(0DH);
1                CALL CO(0AH);
1                    IF(TD<L)THEN GOTO BURNER;   /*IF TD<L & TS<L*/
1                          ELSE GOTO LOOP; /*IF TD>L & TS<L*/

1            MIX:     OUTPUT(0E4H)=(OE4 OR 07H);   /*SELECT AMBIENT
                                TEMP SENSOR*/
1                OUTPUT(0EAH)=1;
1                OUTPUT(0EAH)=0;
1                CALL TIME(100);
1                TA=(0C0H AND INPUT(0EAH))*4+(NOT INPUT(0E9H));

                 /*TA=AMBIENT TEMP*/

1                CALL CO('M');
1                CALL CO('I');
1                CALL CO('X');
1                CALL CO(0AH);
1                CALL CO(0DH);
1                CALL NMOUT(HIGH(TA));
1                CALL NMOUT(LOW(TA));
1                CALL CO(' ');
1                CALL CO('T');
1                CALL CO('A');
1                CALL CO(0DH);
1                CALL CO(0AH);

             /*CHECK DAMPER POSITION; IF FULLY OPEN, THETA-50*/

1                IF((INPUT(0E8H) AND 1)=1)THEN THETA=0;
1                IF((INPUT(0E8H) AND 2)=2)THEN THETA=50;

                 /*ITERATIVE PROCEDURE FOR DETERMINING
                 THEORETICAL SETTING ,GIVEN TS & TD*/

1                N,A=25;
1                ALT=1;
1                CALL CO('I');
1                CALL CO('T');
1                CALL CO('E');
1                CALL CO('R');
1                CALL CO(0DH);
1                CALL CO(0AH);
1            ITER:    TD=(TS*10/KS(N)+TA*10/KA(N))*10;
1                  IF((L<TD) AND (TD<U))THEN DO;
2                              DO WHILE (N<THETA);
3                              CALL BKWD;
```

```
180   3                                   THETA=THETA-1;
181   3                                   END;
182   2                                   DO WHILE(N>THETA);
183   3                                   CALL FWD;
184   3                                   THETA=THETA+1;
185   3                                   END;
186   2                                  END;

187   1                  IF(TD<L)THEN DO;
189   2                          A=(A+ALT)/2;
190   2                          IF(N=23)THEN N=24;
192   2                          N=N+A;
193   2                          ALT=-ALT;
194   2                          END;
195   1                  IF(TD>U)THEN DO;
197   2                          A=(ALT+A)/2;
198   2                          ALT=-ALT;
199   2                          IF(N=27)THEN N=26;
201   2                          N=N-A;
202   2                          END;
203   1                  IF(N<>THETA)THEN GOTO ITER;

              /*THIS ROUTINE WILL MAINTAIN THE CORRECT DAMPER SETTING
                      FOR THE  DESIRED DRYER TEMPERATURE*/

205   1              MAINT:  OF4=(OE4 AND 0F8H) OR 6;
206   1                  OUTPUT(0E4H)=OF4;                          /*SELECT TD*/
207   1                  OUTPUT(0EAH)=1;
208   1                  OUTPUT(0EAH)=0;
209   1                  CALL CO('M');
210   1                  CALL CO('A');
211   1                  CALL CO('I');
212   1                  CALL CO('N');
213   1                  CALL CO('T');
214   1                  CALL CO(0AH);
215   1                  CALL CO(0DH);
216   1                  CALL DELAY;      /*10 SEC DELAY*/

                         /*MEASURE DRYER TEMP*/

217   1                  TD=(0C0H AND INPUT(0EAH))*4+(NOT INPUT(0F8H));
218   1                  CALL NMOUT(HIGH(TD));
219   1                  CALL NMOUT(LOW(TD));

220   1                  CALL CO(' ');
221   1                  CALL CO('T');
222   1                  CALL CO('D');
223   1                  CALL CO(0AH);
224   1                  CALL CO(0DH);

              /*THE FOLLOWING ROUTINES COMPENSATE FOR ANY POSSIBLE INACCURACIES
                  IN THE THEORETICAL CALCULATION FOR THE DAMPER SETTING*/

225   1              DCR:    IF((TD>U) AND (THETA>0)) THEN DO;
227   2                                  THETA=THETA-1;
228   2                                  CALL FWD;
229   2                                  END;
```

```
231   1         INCR:   IF((TD<L) AND (THETA<50)) THEN DO;
232   2                                CALL FWD;
233   2                                THETA=THETA+1;
234   2                                END;
235   1                 OE4=(OE4 AND 0F8H) OR 5;
236   1                 OUTPUT(0E4H)=OE4;          /*SELECT TS*/
237   1                 OUTPUT(0EAH)=1;
238   1                 OUTPUT(0EAH)=0;
239   1                 CALL TIME(100);
240   1                 TS=(0C0H AND INPUT(0EAH))*4+(NOT INPUT(0E8H));
241   1                 CALL NMOUT(HIGH(TS));
242   1                 CALL NMOUT(LOW(TS));
243   1                 CALL CO(' ');
244   1                 CALL CO('T');
245   1                 CALL CO('S');
246   1                 CALL CO(0AH);
247   1                 CALL CO(0DH);

          /*THIS ROUTINE WILL MAINTAIN DESIRED DRYER TEMP INDEFINITELY
              UNLESS SOLAR TEMP DROPS BELOW LOWER LIMIT;
              IN WHICH CASE, PROGRAM RETURNS TO SOLAR ROUTINE*/

248   1                 IF(TS<=L)THEN GOTO SOLAR;

          /*IF THE DRYER COMPARTMENT BECOMES TOO HOT OR TO COOL,
          AND THE DAMPER IS AT MIN OR MAX SETTING RESPECTIVELY.
          AN ALARM LIGHT WILL BE ACTIVATED AND THE PROGRAM HALTED*/

250   1                 IF((TD>U) AND (THETA=0))THEN GO TO ALARM;
252   1                 IF((TD<L) AND (THETA=50))THEN GOTO ALARM;

254   1                 GO TO MAINT;

255   1         ALARM:  OE4=OE4 AND 0FFH;    /*SET ALARM BIT*/
256   1                 OUTPUT(0E4H)=OE4;    /*RAISE ALARM LINE*/
257   1         END USDA;


MODULE INFORMATION:

     CODE AREA SIZE     = 0700H   1792D
     VARIABLE AREA SIZE = 0011H     17D
     MAXIMUM STACK SIZE = 0006H      6D
     315 LINES READ
     0 PROGRAM ERROR(S)

END OF PL/M-80 COMPILATION
```

E-21-662

FINAL REPORT

CONTROL SYSTEMS FOR INTERFACING SOLAR AND BIOMASS FUELS IN AGRICULTURAL DRYERS

Principal Investigator:

Dr. J. H. Schlag
Professor

Project Manager:

Dr. A. P. Sheppard
Acting Vice President for Research

Submitted to the

U.S. DEPARTMENT OF ENERGY

through the

U.S. DEPARTMENT OF AGRICULTURE

GEORGIA INSTITUTE OF TECHNOLOGY
SCHOOL OF ELECTRICAL ENGINEERING
ATLANTA, GEORGIA 30332

FINAL REPORT

CONTROL SYSTEMS FOR INTERFACING SOLAR AND BIOMASS FUELS IN AGRICULTURAL DRYERS

Submitted to the
U.S. DEPARTMENT OF ENERGY

through the
U.S. DEPARTMENT OF AGRICULTURE

Principal Investigator:
Dr. J. H. Schlag
Professor

Project Manager:
Dr. A. P. Sheppard
Acting Vice President for Research

CONTROL SYSTEMS
FOR INTERFACING SOLAR AND BIOMASS FUELS
IN AGRICULTURAL DRYERS

AUTHORS:

J. H. SCHLAG

A. P. SHEPPARD

I. G. WELLS

W. E. BAIRD

CONTRIBUTOR:

R. S. ZIMLER

TABLE OF CONTENTS

TABLE OF FIGURES

TABLE OF FIGURES (CONTINUED)

I. INTRODUCTION

Previous work has shown that solar energy is a feasible alternative to fossil fuels for heating air used in crop drying. However, the greatest drawback to solar energy is that it is not continuously available, and other energy sources must be used as backup. Research is being conducted into the use of biomass to generate methane and alcohol which could be used as backup fuels for solar dryers. In order to interface direct solar-heated air, ambient air, and auxiliary biomass-fueled heat, some means of intelligent process control is required for optimum energy performance. At one time this would have necessitated access to a large computing facility, having many capabilities in addition to the control function needed. Now, however, microprocessor technology has provided smaller units of computing power so that a relatively inexpensive microprocessor of more limited ability can be dedicated to a single task. The application of a microprocessor to the interfacing problem, together with modeling and construction of the air-mixing system, was the overall objective of this work.

There are several reasons why solar energy and intelligent microprocessor control represent important innovations for agricultural drying. The need for abundant, inexpensive energy for drying is demonstrated in several ways. The largest quantities of energy for agricultural drying in Georgia and the

Southeast are presently consumed on the peanut and tobacco crops, with peanuts valuable both as a direct source of protein and indirectly as a feed for domestic animals. In addition, growing demand for forage crops to replace grain as food for cattle will result in expanded drying operations. As continuing growth in the world population causes increased demand for low-cost protein, the importance of these feed crops especially suited to the southeastern United States will rise. Thus, the production of peanuts, tobacco, and forage crops alone would be significantly assisted by the low-cost availability of solar-powered dryers.

Furthermore, the detailed requirements of optimum crop drying indicate the careful monitoring of temperature that a microprocessor control system can provide. For example, peanuts must be dried quickly enough to avoid the growth of mold, one type of which produces a rather toxic carcinogen called aflatoxin. The aflavus mold enters any nut with cracks and grows favorably at about 77°F. If the moisture level is reduced below 10%, the mold will not form. On the other hand, peanuts dried at temperatures much above 90°F in conventional dryers become bitter and bring lower market prices. For these reasons, drying chamber temperatures for peanuts must be carefully controlled.

Another example of the complexity of the drying process is shown by tobacco, one of the three largest cash crops and the largest user of agricultural process energy in Georgia. It requires substantial heat input for curing and drying with as much as 400-500% of the final tobacco weight being water before curing. Tobacco curing is additionally complicated because the leaves are "living" matter when the process begins and require careful attention if the final cured product is to be of high quality.

Finally, even the drying of forage should be performed in a temperature-controlled setting, so that nutritional value is maximized and oxidation and browning minimized. Thus the various requirements which must be met in agricultural drying indicate the need for alternative energy sources and intelligent automatic control, as provided by a microprocessor system.

II. THE MODEL

A. Introduction

The work on this project began with the design of a mathematical model of the drying process; the primary source of heat energy was taken to be solar-heated air supplied by conventional collectors. The temperature of this solar-heated air would be lowered, when necessary, by mixing with ambient air; it would be raised, when necessary, by the addition of biomass-fueled heat energy. The microprocessor would receive measurements of various temperatures and use these values to make decisions governing the operational modes of the drying system. Parts of the model would thus be implemented in developing the algorithms used by the microprocessor in the decision process. After construction of the prototype system, partial verification of the model was performed experimentally.

The process control system is shown in Figure 1. The valves to ambient air and the solar collectors are controlled by the microprocessor, as is the biomass gas inlet. The inputs to the microprocessor are ambient air temperature, solar collector temperature, and drying chamber temperature; the outputs are the controls on the valve openings and gas inlet. The model which was developed is essentially a static description of the relationships involved. However, because of the large time-span involved in drying crops (of the order of 24 hours)

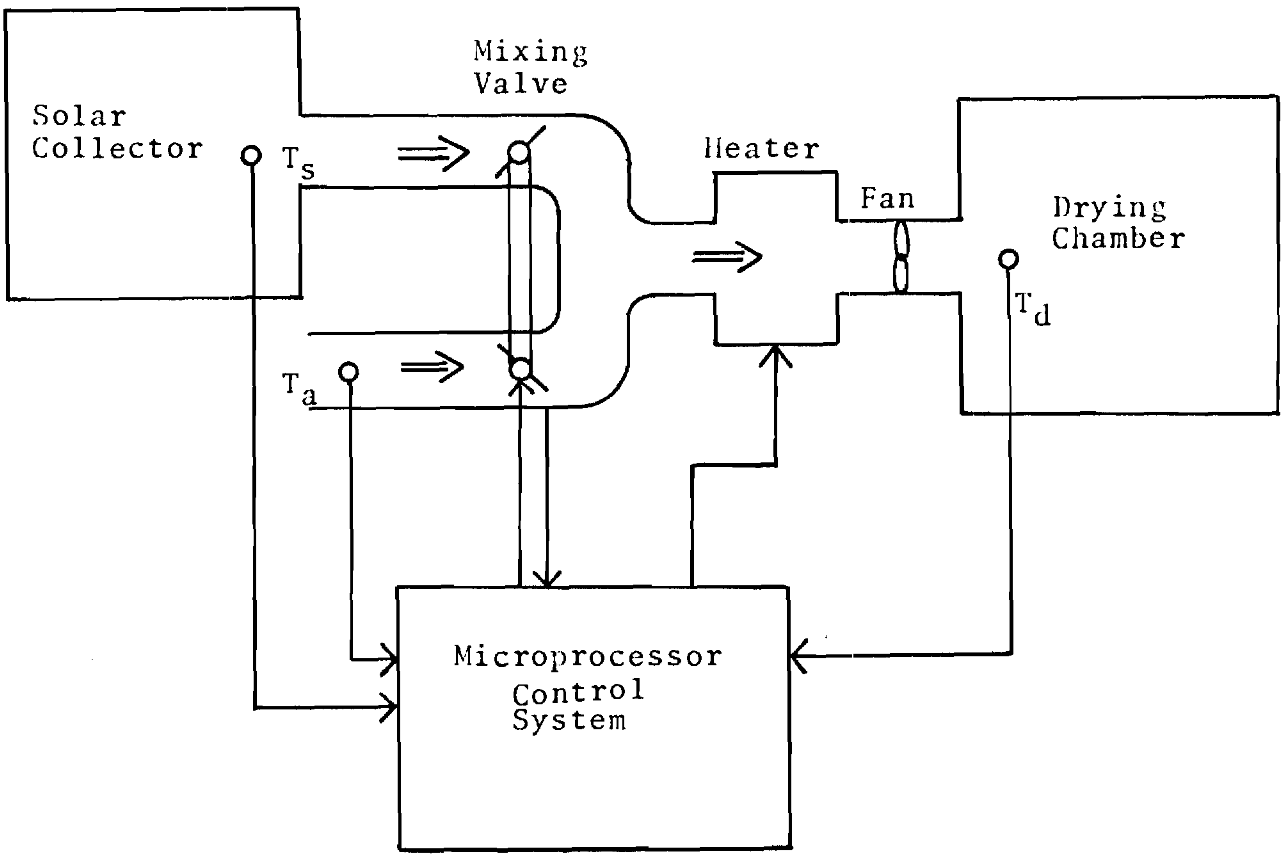


FIGURE 1. Process Control System

and the slow response time of valve mechanisms, static relationships can be employed by the microprocessor in an iterative manner, effectively producing a slow-dynamic model.

The first question addressed by the model is how the ideal drying temperature for a particular substance determines the length of time required to bring the moisture content from some initial level to a predetermined final level.

The second question addressed is what combination of ambient air, solar-heated air, and additional biomass thermal energy is most appropriate and economical under prevailing climatic conditions, i.e., ambient temperature, insolation, etc.

Finally the desired rates of flow of ambient air and solar-heated air must be related to the angular openings of the two valves and to any resistance to air flow (drag) inherent in the two sources.

B. Definition of Symbols

Let symbols be defined as follows:

Q_{req} = heat energy required for drying

Q_{sup} = heat energy supplied by ambient air, solar collectors, biomass fuel, or some combination of these three

m = mass of matter to be dried

M = mass of air used in drying

u_i = initial water fraction in matter to be dried

u_f = final water fraction in matter to be dried

c_w = specific heat capacity of water = 1.0 Kcal/KgC^o

c_a = specific heat capacity of ambient air

h_v = heat of vaporization of water at the drying temperature

h_c = heat of combustion of biomass fuel

T_a = temperature (^{o}C) of ambient air

T_s = temperature of solar collector system

T_d = ideal drying temperature for a particular type of matter

T_m = initial temperature of matter to be dried

T_i = temperature of warm air before heating by biomass fuel

p_a = density of ambient air

$\dot{v}_d$ = total volume flow rate provided by fan, assumed constant

$\dot{v}_a$ = volume flow rate of ambient air

$\dot{v}_s$ = volume flow rate of solar-heated air

Dt = time duration of drying process

$\dot{q}_{req}$ = thermal power required from biomass fuel

$\dot{q}_{sup}$ = thermal power supplied by biomass fuel

$\dot{v}$ = volume flow rate for gas in the biomass heater

θ = angle of opening of valve to solar collector system

ϕ = angle of opening of valve to ambient air

r = radius of duct pipe

C. Explanation of Model

From a consideration of the expressions for Q_{req} and Q_{sup} the relation between T_d and Dt can be shown:

$$Q_{req} = (u_i - u_f)m\, c_w(T_d - T_m) + (u_i - u_f)m\, h_v \tag{1}$$

$$\begin{aligned} Q_{sup} &= M\, c_a T_d \\ &= p_a\, \dot{v}_d\, Dt\, c_a T_d \end{aligned} \tag{2}$$

By recognizing that Q_{sup} must equal Q_{req}, and that p_a, $\dot{v}_d$, and c_a are not adjustable, it follows that T_d determines Dt.

The discussion now concerns how usage of ambient air, solar-heated air and biomass fuel is determined.

Case 1: If $T_a < T_d$ and $T_s = T_d$, then open only the valve to the solar collectors, leaving valve to ambient air closed and biomass heater off. In this case $\dot{v}_s = \dot{v}_d$, and $\dot{v}_a = 0$.

Case 2: If $T_a < T_d$ and $T_s > T_d$, then leave biomass heater off and open both valves (to ambient air and solar collectors) to appropriate degrees so that

$$\begin{cases} \dot{v}_a + \dot{v}_s = \dot{v}_d \\ T_a\dot{v}_a + T_s\dot{v}_s = T_d\dot{v}_d \end{cases} \tag{3}$$

whose solution is

$$\dot{v}_a = \dot{v}_d\left(\frac{T_s - T_d}{T_s - T_a}\right) \qquad \dot{v}_s = \dot{v}_d\left(\frac{T_d - T_a}{T_s - T_a}\right) \tag{4}$$

Case 3: Finally, if $T_a < T_d$ and $T_s < T_d$, then extra heat energy will have to be added by the biomass-powered heater. In this case:

$$\dot{v}_a + \dot{v}_s = \dot{v}_d$$

$$T_a\dot{v}_a + T_s\dot{v}_s = T_i\dot{v}_d < T_d\dot{v}_d \tag{5}$$

and $T_i < T_d$

In order to raise T_i to T_d, $\dot{q}_{req}$ Kcal/sec must be added, where

$$\dot{q}_{reg} = \frac{d}{dt}\left(M\ c_a(T_d - T_i)\right)$$

$$= c_a(T_d - T_i)\ \rho_a\ \dot{v}_d \tag{6}$$

The thermal power supplied is given by

$$\dot{q}_{sup} = h_c\ \dot{v} \tag{7}$$

Since $\dot{q}_{sup}$ must equal $\dot{q}_{req}$, T_i determines $\dot{v}$.

Although there will be a thermocouple positioned to measure the temperature T_i of the air mixed from the ambient air valve and the solar system valve, it is a simple matter to calculate it from T_s, T_a, $\dot{v}_a$, and $\dot{v}_s$.

$$T_i = \frac{T_a \dot{v}_a + T_s \dot{v}_s}{\dot{v}_a + \dot{v}_s} \tag{8}$$

Even though it is intuitively apparent in this form as a weighted average, the expression can also be found by using a simple heat transfer calculation.

It remains to discuss how the volume flow rate for a valve depends on angular opening and on any resistive structures within. When the two valves are used together, these flow rates will be coupled, as stated earlier.

$$\dot{v}_a + \dot{v}_s = \dot{v}_d$$

It will be useful to consider an analogous electrical circuit. Let the resistance within the ambient air system be R_a, that within the solar system be R_s. Let the resistance offered by the solar system valve (open to angle θ) be R_θ, that offered by the ambient air valve (open to angle ϕ) be R_ϕ. Let the total air flow rate $\dot{v}_d$ be represented by the total current in the circuit i_d, and the flow rates in the "parallel" valves by i_a and i_s. The source of emf (pressure) is E. A diagram of the circuit is given in Figure 2. The solutions to this circuit for $i_a(\dot{v}_a)$ and $i_s(\dot{v}_s)$ are given by:

$$i_a = i_d \left(\frac{R_s + R_\theta}{R_a + R_\phi + R_s + R_\theta}\right) \tag{9a}$$

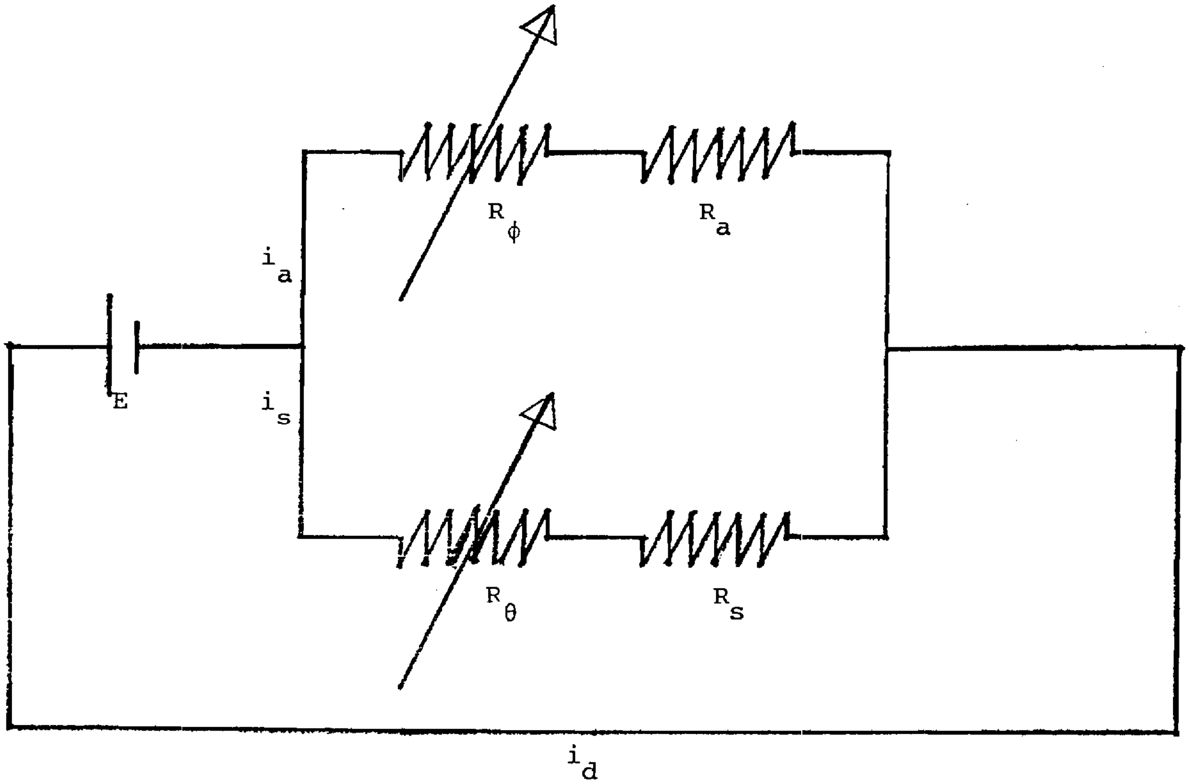


FIGURE 2

CIRCUIT DIAGRAM

$$i_s = i_d \left(\frac{R_a + R_\phi}{R_a + R_\phi + R_s + R_\theta}\right) \tag{9b}$$

It is clear from the solutions to this analogous electrical circuit that the drag factors in the two air paths of the drying system would be closely coupled. Since there is no a priori way of modeling the air drag quantities represented by R_a, R_ϕ, R_s, and R_θ, empirical measurements would be used to obtain their relative magnitudes. One useful relationship can be obtained, however. It seems reasonable that the amount of drag presented to the air flow by a valve would be proportional to the projected area of the valve on the plane of the opening; thus

$$R_\phi \propto \pi r^2 \cos\phi \quad \text{and} \quad R_\theta \propto \pi r^2 \cos\theta \tag{10}$$

where it is assumed that the radius of the duct pipe r will be the same throughout the system. Nevertheless, an exact understanding of the magnitudes of these four drag factors will not be necessary because of the iterative control procedure employed: an initial setting of the air valves and gas inlet will be determined by the input parameters; the intermediate temperature and ultimate drying temperature will be checked; the air valves and gas inlet will be re-set, if necessary; then temperatures will be re-checked and the iteration process continued.

It will be shown in subsequent sections of this report that only certain parts of this model were needed in the microprocessor algorithm and that certain adaptations were made in order to facilitate programming. Also, the postulated cosine dependence of flow resistance (drag) with respect to angle of opening was tested experimentally, with graphical comparison of results.

III. THE MICROPROCESSOR CONTROL SYSTEM

The microprocessor based control system, as shown in Figure 3, contains an Intel 80/20 microprocessor, an Intersil 7901 A/D converter, three LM3911 temperature sensors, a Slo-Syn stepping motor, and a Slo-Syn stepping motor controller. The three temperature transducers are used to measure the solar temperature, T_s, the ambient temperature, T_a, and the drying temperature, T_d. These three temperatures are selected by the microprocessor through the multiplexer, and converted to digital form by the A/D converter.

The microprocessor then compares the three temperatures and decides whether to increase or decrease the relative amount of solar versus ambient air. The microprocessor increases or decreases the relative flow by sending pulses and direction to the stepping motor controller, which, in turn, sends the stepping motor in either a clockwise or a counterclockwise direction depending on the direction level from the microprocessor. The stepping motor is connected to the vanes in the flow of solar and ambient air, so that as the solar vanes are opened the ambient air vanes are closed, and vice versa.

A brief description of each component of the system follows. The LM3911 temperature transducers are commercially available integrated circuits that develop a voltage linearly proportional to temperature in degrees Kelvin. They have an approximate

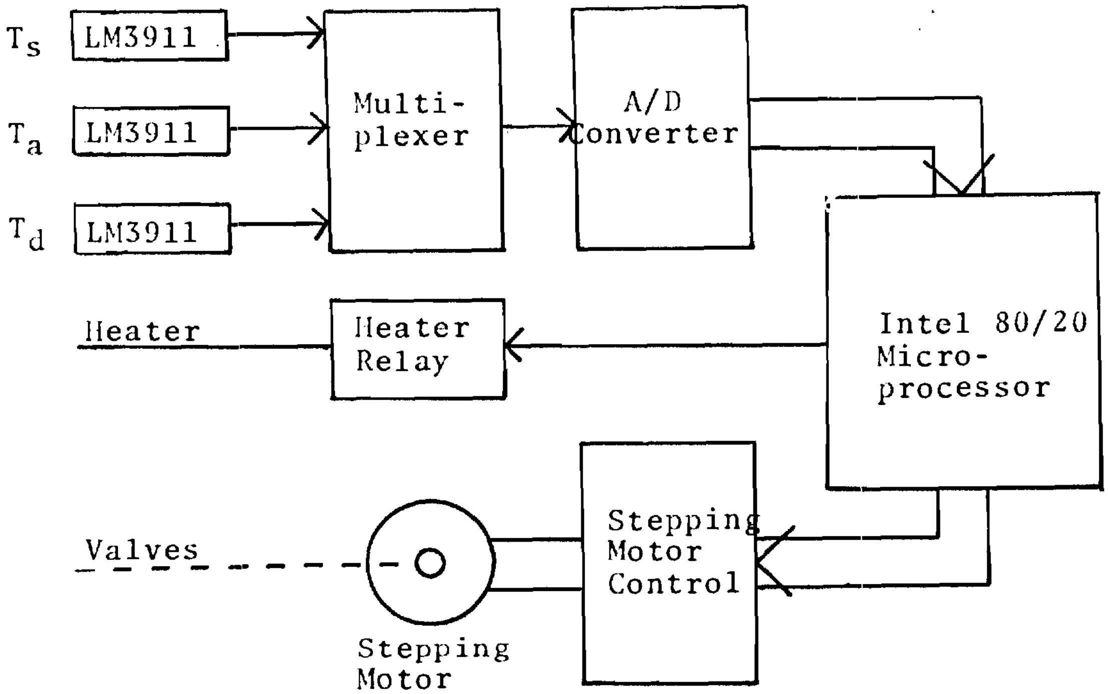


FIGURE 3. Microprocessor Control System

sensitivity of 10 mV/^{o}K over an operating temperature range from 248^{o}K to 358^{o}K or -25^{o}C to +85^{o}C. The outputs from the temperature transducers are converted from degrees Kelvin to degrees Centigrade by using a simple offset operational amplifier circuit with an LM308 operational amplifier.

The multiplexer is a Datel model #MM-8, solid state multiplexer using FET switches and integral TTL-compatible decoding. This unit can be connected directly to the output of the Intel microprocessor, so that any one of the three temperature transducers can be selected by a binary code sent from the microprocessor.

The analog-to-digital (A/D) converter is used to convert the output from the multiplexer into a binary form for input to the microprocessor system. This unit has 12 bits of output and has an accuracy of ±1 mV.

The microprocessor system is an Intel SBC 80/20, single board computer with an 8080 microprocessor, two kilobytes of random-access memory (RAM), four kilobytes of EPROM memory, 48 lines of digital input or output, asynchronous serial I/O port, and a real time clock. The EPROM memory is used to hold the program that executes the control algorithm on the 8080 microprocessor. The random-access memory is used as a "scratch pad" to hold data for the control algorithm. Digital output lines are used to control the multiplexer, A/D conversion, and stepping motor control. Digital inputs are used for the input of temperature data. The serial I/O board is used in conjunction with a CRT terminal to print out the status of the controller algorithm at points for debugging and diagnostic purposes. The real time clock is used for generating real time delays in the control algorithm program.

The Slo-Syn model #M092-FD09 stepping motor and the Slo-Syn model #STM1800 stepping motor controller are used to control the opening and closing of the solar air vane and the ambient temperature air vane. The stepping motor can be controlled in incremental 1.8^{o} steps throughout its revolution, giving 200 steps per revolution. The motor has a maximum torque of 450 oz-inch and a maximum speed of 1,000 steps/second. The stepping motor controller contains all the electronics necessary to drive the stepping motor windings and requires only a direction and step pulse from the microprocessor. The direction input is a TTL level from the microprocessor to control clockwise or counterclockwise movement of the armature. The step input requires a TTL level pulse, with each pulse corresponding to a 1.8^{o} movement of the motor. The motor and motor controller will maintain torque to hold the present position of the motor between control pulses from the microprocessor.

IV. THE CONTROL ALGORITHM

The microprocessor uses a combination of both a feed-forward control algorithm and a feed-back control algorithm. The feed-forward algorithm offers the advantage of very rapid response to sudden input changes with no stability problems. The feed-back scheme offers the advantage of high accuracy and adaptation to changes in system parameters. Because the microprocessor is an intelligent controller, it can be programmed to take advantage of both types of control schemes, using the feed-forward scheme for rapid changes, and then switching to the feed-back scheme for maintaining a precise drying temperature.

The feed-forward control scheme looks at the solar air temperature and the ambient air temperature and calculates the position of the motor control on the valves, based on the model of the mixing system discussed in Section II. This model was developed by assuming a simple cosine relationship between valve angle and resistance to airflow in the air inputs as shown in equation (10), section II.

$$R_\phi \propto \pi r^2 \cos \phi \qquad\qquad R_\theta \propto \pi r^2 \cos \theta \tag{1}$$

Using the results given for the analogous electric circuit (equation (9), section II), and replacing the currents i_a, i_s, and i_d with the corresponding volume flow rates $\dot{v}_a$, $\dot{v}_s$, and $\dot{v}_d$, we have

$$\dot{v}_a = \dot{v}_d \left(\frac{R_s + R_\theta}{R_a + R_\phi + R_s + R_\theta}\right) \tag{2}$$

$$\dot{v}_s = \dot{v}_d \left(\frac{R_a + R_\phi}{R_a + R_\phi + R_s + R_\theta}\right) \tag{3}$$

where

$\dot{v}_a$ = volume rate of flow for ambient air

$\dot{v}_s$ = volume rate of flow for solar-heated air

$\dot{v}_d$ = total volume rate of flow in drying chamber

R_ϕ = resistance offered by ambient air valve (open to angle ϕ)

R_a = other resistance within ambient air path

R_θ = resistance offered by solar air valve (open to angle θ)

R_s = other resistance within solar air path

Because the laboratory prototype uses an electric heating coil, occupying negligible space in the duct pipe, to simulate the solar-heated air, the following assumption is also appropriate:

$$R_a = R_s = 0 \tag{4}$$

Also, the ambient air valve and the solar valve are operated simultaneously by the single stepping motor so that we have

$$\phi = 90^\circ - \theta \tag{5}$$

Using equations (1), (4), and (5) in equations (2) and (3), we obtain

$$\dot{V}_a = \dot{V}_d \left(\frac{\cos\theta}{\sin\theta + \cos\theta}\right) \tag{6}$$

$$\dot{V}_s = \dot{V}_d \left(\frac{\sin\theta}{\sin\theta + \cos\theta}\right) \tag{7}$$

Experimental verification of equation (6) (for the ambient air volume flow rate) is shown in Figure 4, which indicates a model error of approximately 5% over the operating range. Similar agreement was found in the case of the solar air volume flow rate. The resulting solar valve angle as a function of solar temperature, ambient temperature, and desired dryer temperature is given by

$$\theta_d = \arctan\left(\frac{T_d - T_a}{T_s - T_d}\right) \tag{8}$$

The structure of the control algorithm is as follows:

1.) measure T_s and T_a

2.) calculate desired valve angle θ_d

3.) calculate the desired angle change by subtracting the present angle, θ_p (known from the last iteration of the control algorithm), from the desired angle

4.) determine the direction level to the stepping motor controller by the sign (±) of $\theta_d - \theta_p$

5.) determine the number of steps to be sent to the stepping

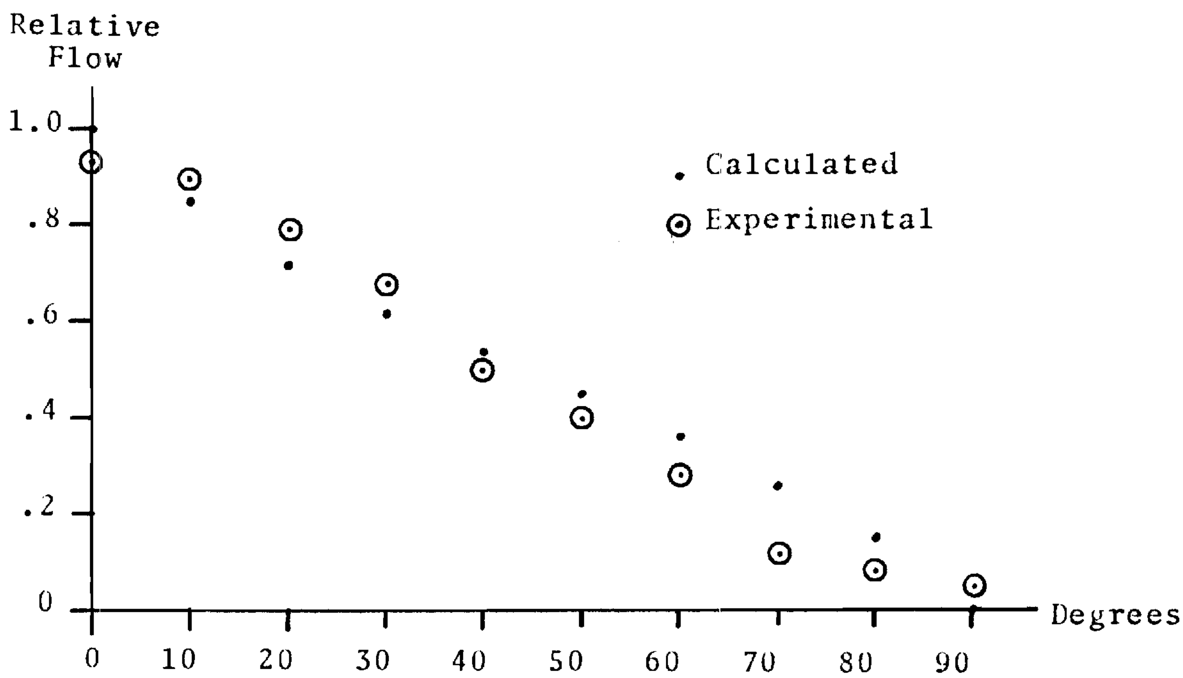


FIGURE 4. Experimental vs. Calculated Air Flow

motor by the magnitude of the difference between θ_d and θ_p. One step would be sent for every 1.8^o difference.

After the algorithm has opened the valve to the projected angle, θ_d, the microprocessor will switch over to the feed-back control scheme for final adjustment of valve angle.

The feed-back control scheme monitors the dryer temperature, T_d, and compares the measured value with the desired temperature value and increases or decreases the valve angle to increase or decrease T_d. This feed-back loop has ten seconds of delay to eliminate the problem of rapid changes in valve position due to feed-back instability. The microprocessor stays in the feed-back algorithm until a sudden change is detected in T_s, which indicates that a more rapid correction is required. The algorithm then reverts back to the feed-forward control scheme.

A third algorithm is used to control the action of a backup methane or propane heater. In this algorithm, T_s is constantly monitored to see if it is below the lower limit of the desired dryer temperature. When T_s falls below the lower limit of the dryer temperature, the gas burner is turned on and the system reverts to a burner algorithm, which simply turns on and off a gas burner to maintain the desired dryer temperature.

V. SOFTWARE

The software program controller is written in PLM/80, which is a high level language which generates the 8080 machine code. Since the 80/20 single board computer is not large enough to run a PLM compiler, the programming is done on an MDS 230 system and the programs are transferred to the 80/20 for execution. Use of the MDS development system and the PLM compiler greatly increases the programming efficiency as well as producing well-documented programs that are easy to understand and modify.

The program flowchart shown in Figure 5 contains three main branches depending on whether the solar air temperature is greater than, less than, or equal to the desired dryer temperature range. If the solar temperature is within the desired dryer temperature range, the right hand loop is used and the program opens the solar valve full and uses the solar air directly for drying.

If the solar air drops below the desired dryer range, the program switches to the left loop which controls the action of the backup burner by turning it on and off to maintain a desired temperature for the dryer. This loop also checks the solar temperature to determine whether it has risen enough to switch back from the backup burner to solar drying.

If the solar temperature gets above the desired drying range, then the middle loop is entered which mixes the solar air with ambient air to achieve a lower desired drying temperature. This

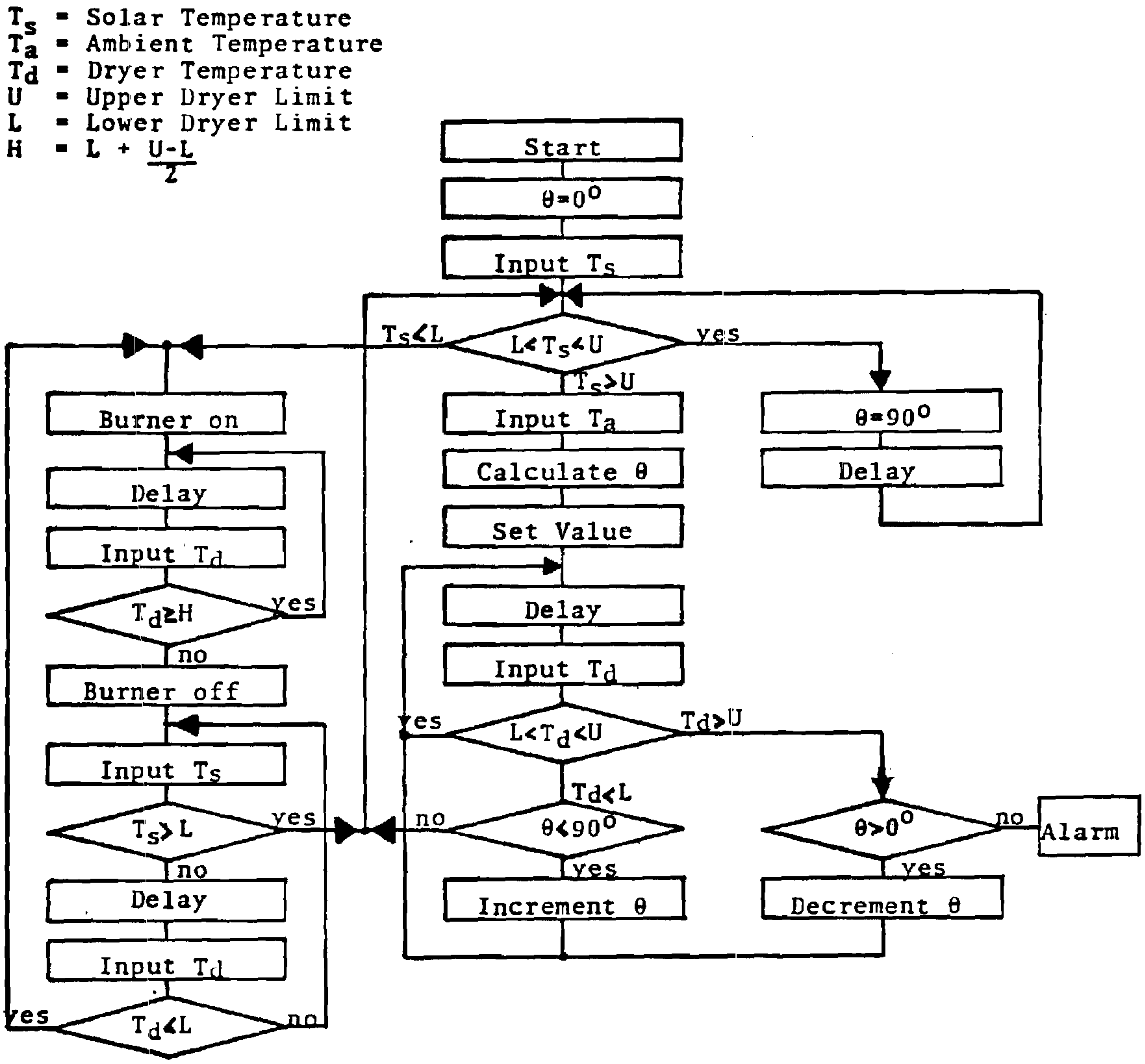


FIGURE 5. Controller Flow Chart

is accomplished by first going through a feed-forward control algorithm where T_s, T_d, and T_a are used to calculate the proper opening for the mixing valves. Once this initial valve opening is achieved, the program reverts to a slower acting feed-back loop to increment or decrement the valve opening to achieve an accurate desired temperature.

VI. EXPERIMENTAL PROCEDURE

To test the operation of the microprocessor controller, an experimental air mixer was constructed using an eight inch axial blower fan and standard eight inch furnace duct. The air mixing valves were constructed by means of a circular louver which rotates inside the duct pipe.

One valve was mounted in the solar air duct and the second valve in the ambient air duct with a chain drive between the valves so that as one valve is opened, the other is closed. An angle of $\theta = 0^o$ represents the solar air being totally shut off and the ambient air being totally open, whereas an angle of $\theta = 90^o$ indicates the solar air valve totally open and the ambient air valve totally closed.

The backup gas heater was constructed from parts of a standard home gas heater system plus a RobertShaw model #VX700 electronic starter system for automatic pilot re-lighting. This system uses a high voltage spark to ignite a pilot light, and the pilot light is then used to light the main burner.

A three phase 3,000 watt electric heater was constructed using heating coils mounted in a section of eight inch duct pipe to simulate the solar air that would normally be available from a standard solar collector. A three phase Variac was used to vary the amount of heat available in the simulated solar collector. By varying the upper and lower desired dryer temperature limits and by changing the amount of power delivered to the solar air simulator, all of the software loops of the controller were checked for proper operation.

Figure 6 shows a plot of solar temperature, T_s, dryer temperature, T_d, and solar louver angle θ as functions of time in a typical test of the feed-back control loop section of the controller program. (During the start-up portion of the graph, the system was initialized using the feed-forward control.) When the dryer temperatures were below the lower desired limit, the controller increased the angle θ to allow more solar heated air. Likewise, when the dryer exceeded the upper desired limit, the controller would decrease the valve angle, reduce the amount of solar heated air, and increase the amount of ambient air. While the system was in the burner control loop of the program, readings were taken of dryer temperature T_d and state of the burner (on-off). This relationship is shown in Figure 7. When T_d exceeds the desired range, the burner turns off; when T_d is below the desired range, the burner is activated. As the graph shows, T_d is not so smoothly level as in Figure 6. This results from the simple two-state configuration of the burner, by contrast with the continuous range of settings of the solar and ambient air valves. Also included is a graph (Figure 8) of the burner temperature response as a function of elapsed time.

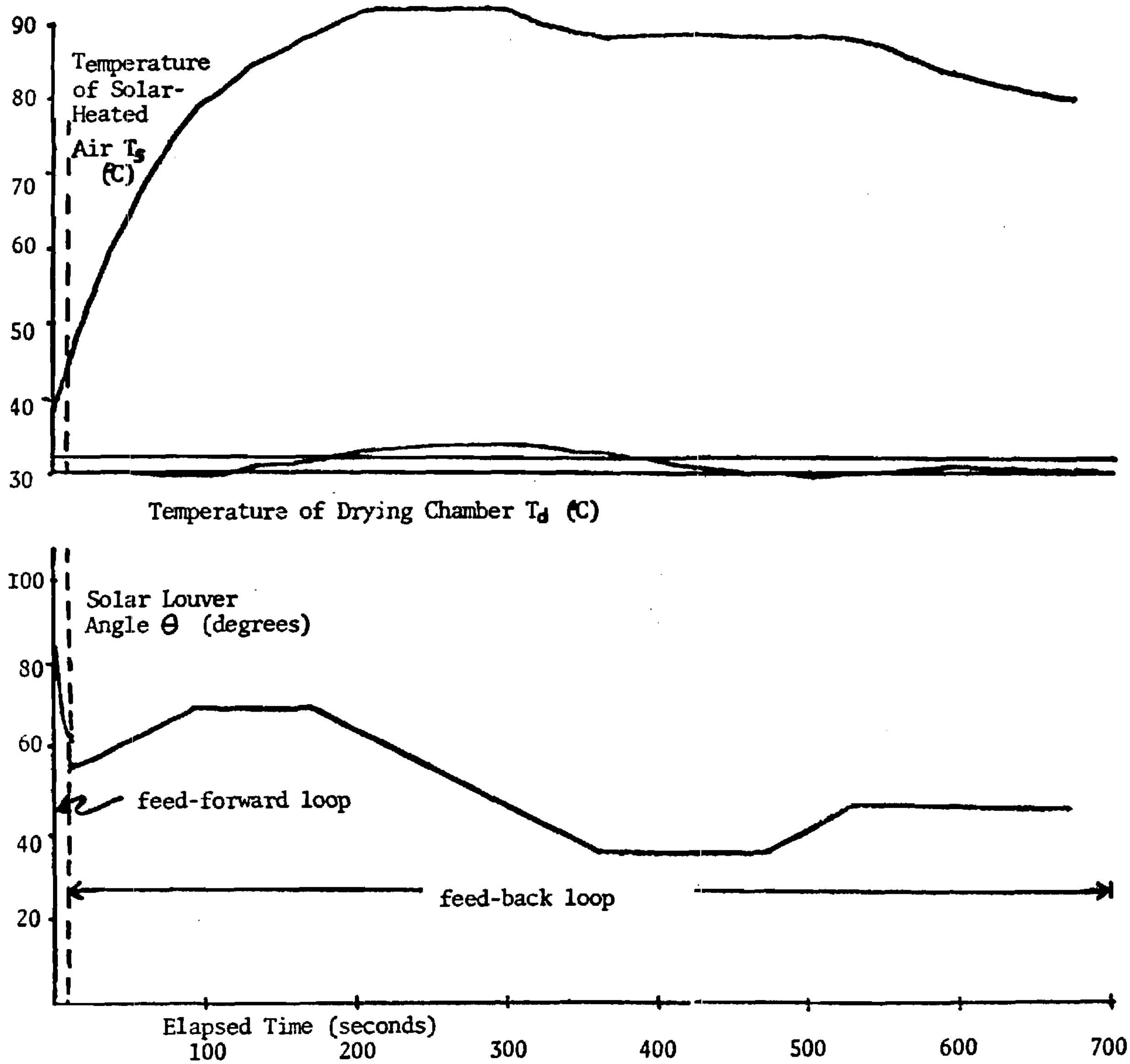


FIGURE 6. T_s, T_d, and θ as Functions of Time During a Test of the Controller Program

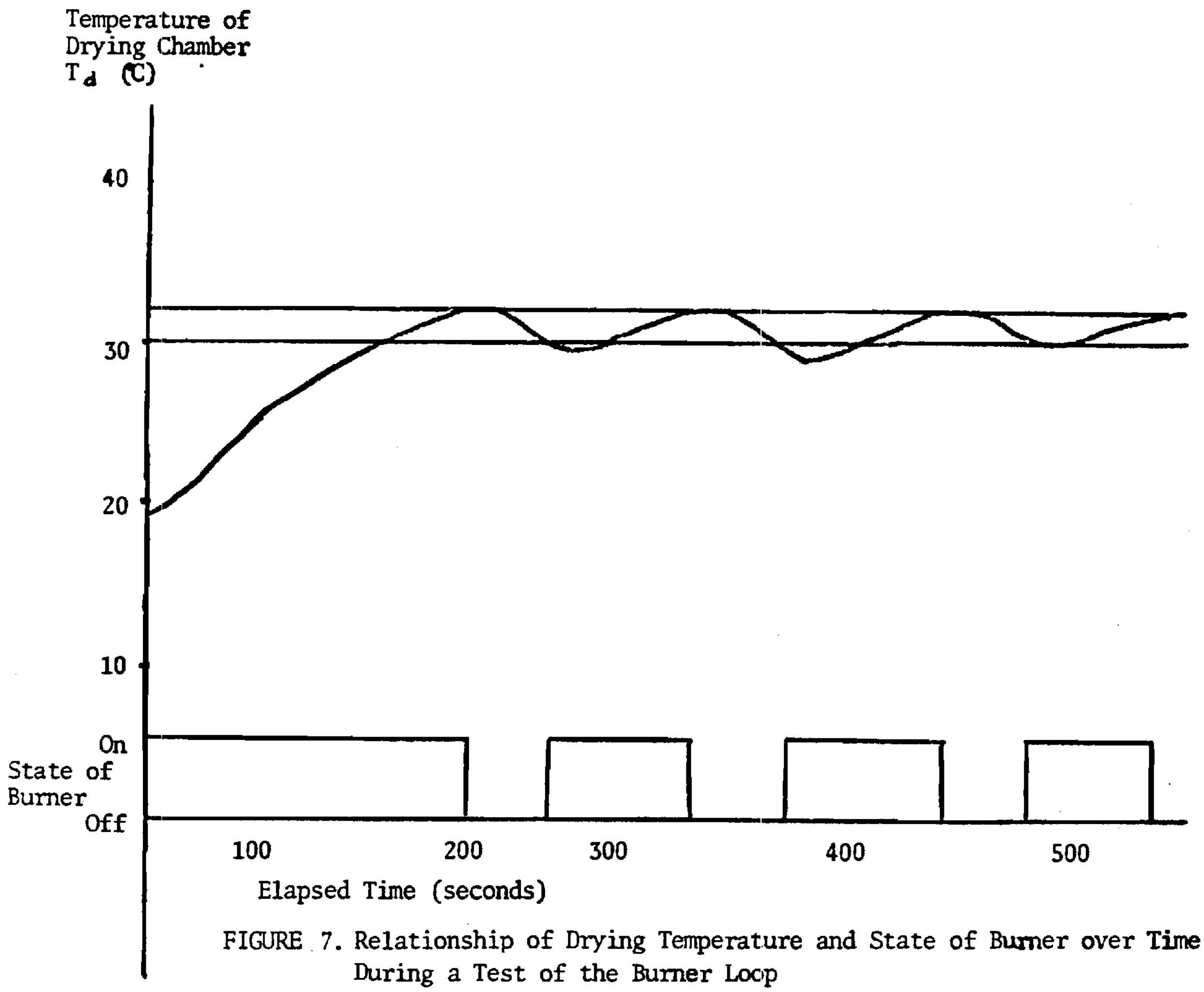


FIGURE 7. Relationship of Drying Temperature and State of Burner over Time During a Test of the Burner Loop

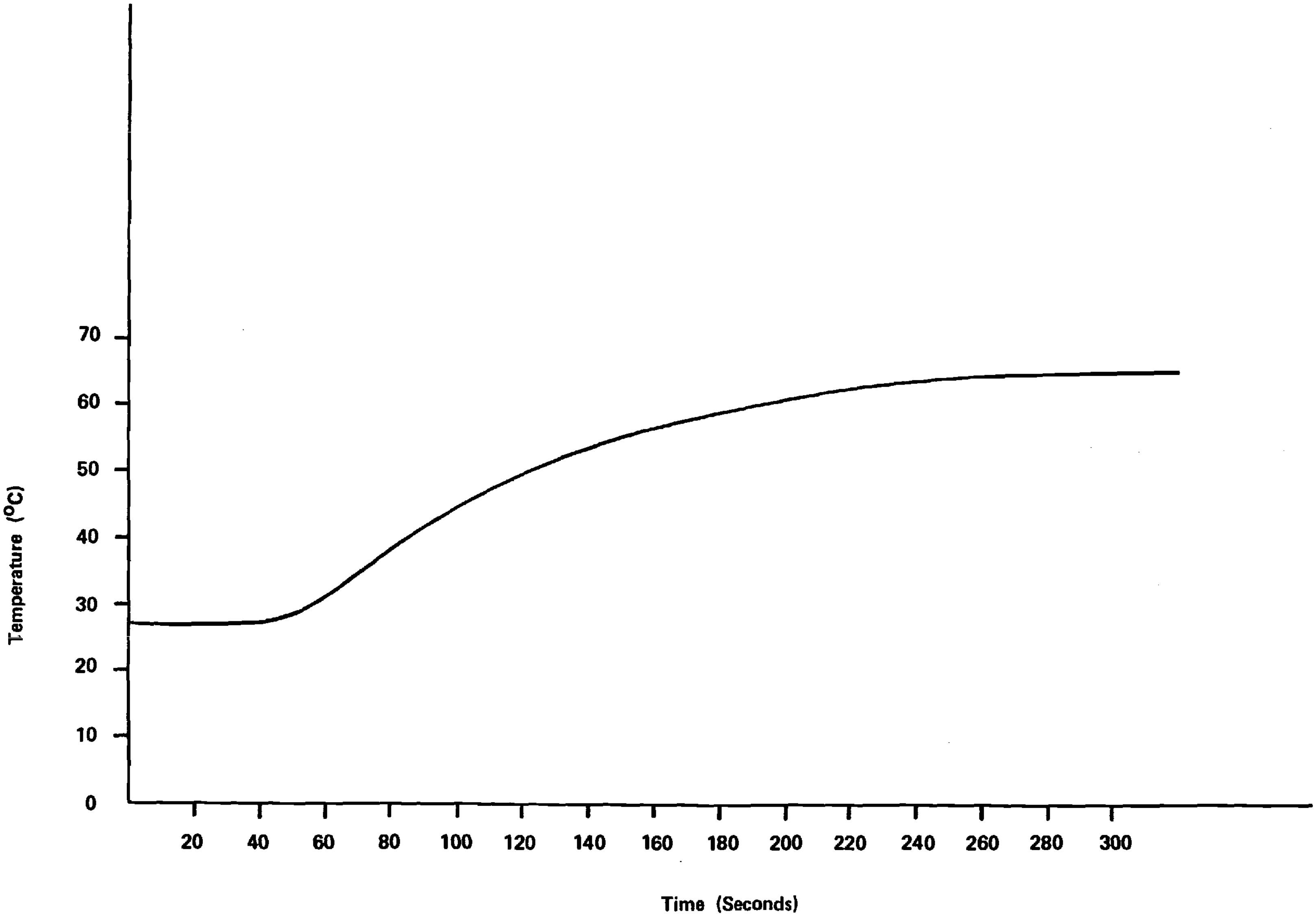


Figure 8. Temperature of Burner–Heated Air vs. Time from Switch–On.

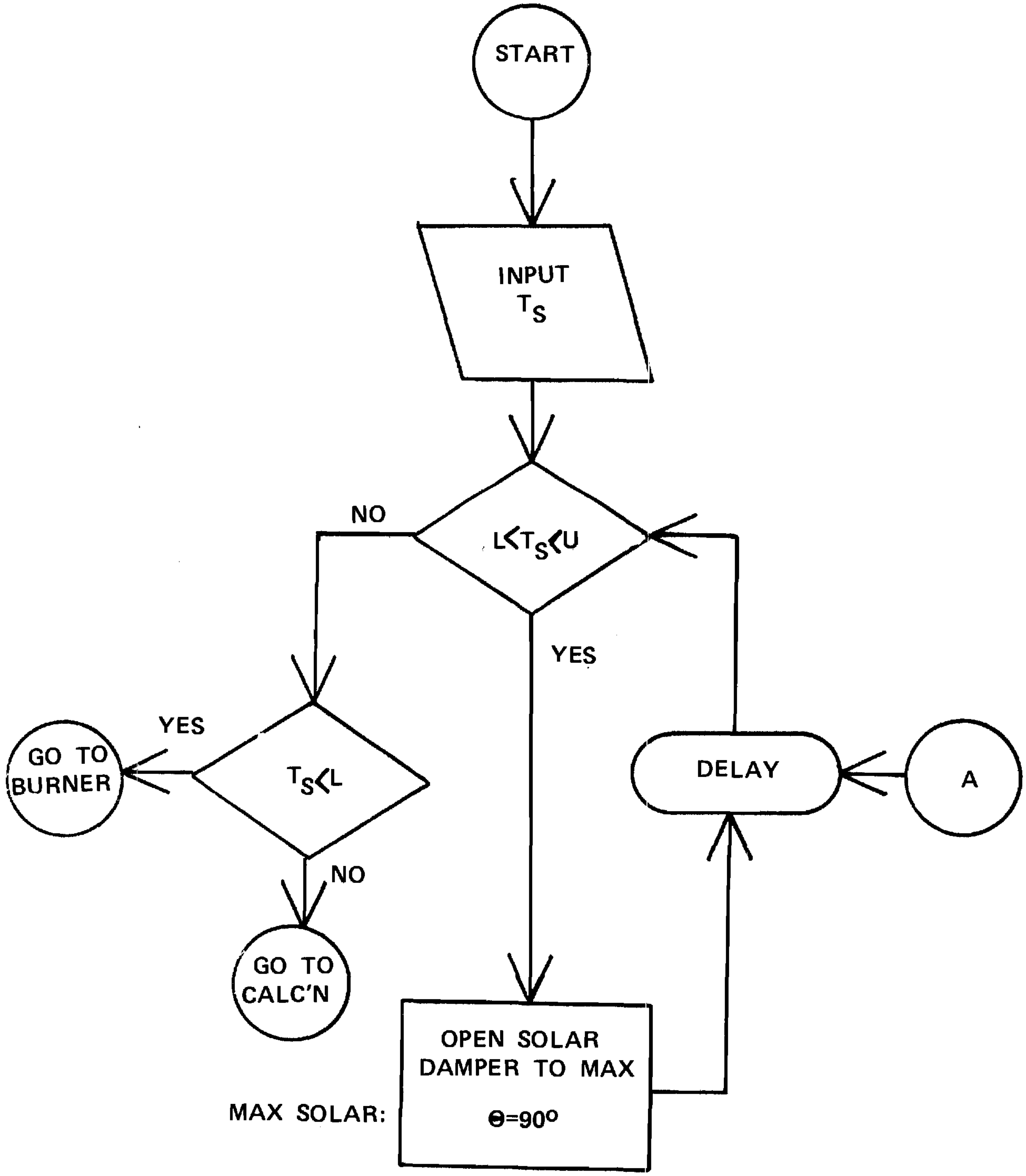
VII. RESULTS AND CONCLUSIONS

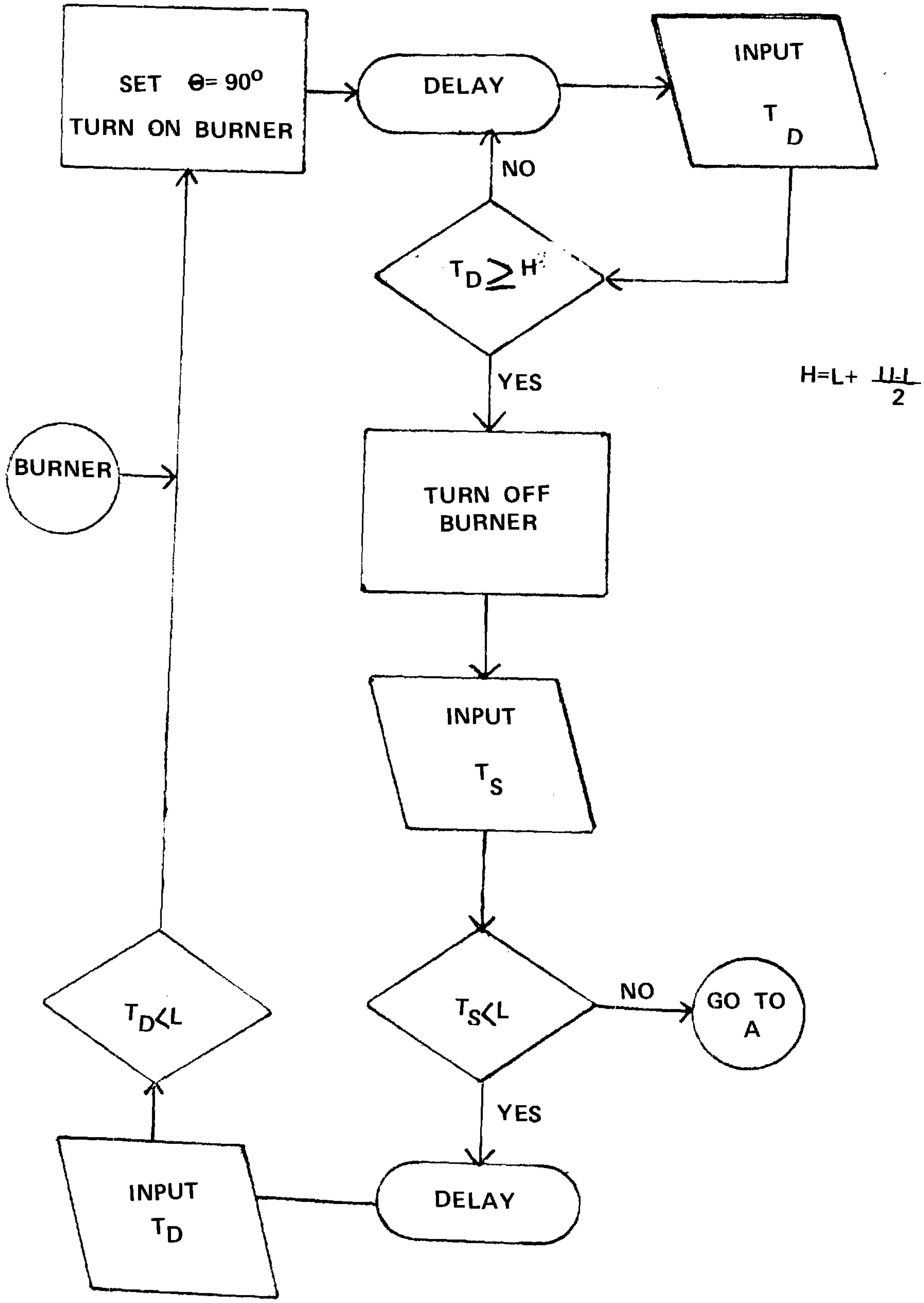
The results of this research indicate that the microprocessor system described is capable of controlling a typical drying process. Since the laboratory equipment cost was less than $2,000, a cost effective system should be feasible on an industrial scale. One of the most important features of the microprocessor system is the ability of the system to expand with little increase in system cost. This feature will permit additional energy conservation algorithms to be incorporated to increase the system cost effectiveness. Additional control functions such as graduated heater control could also be added with little additional cost.

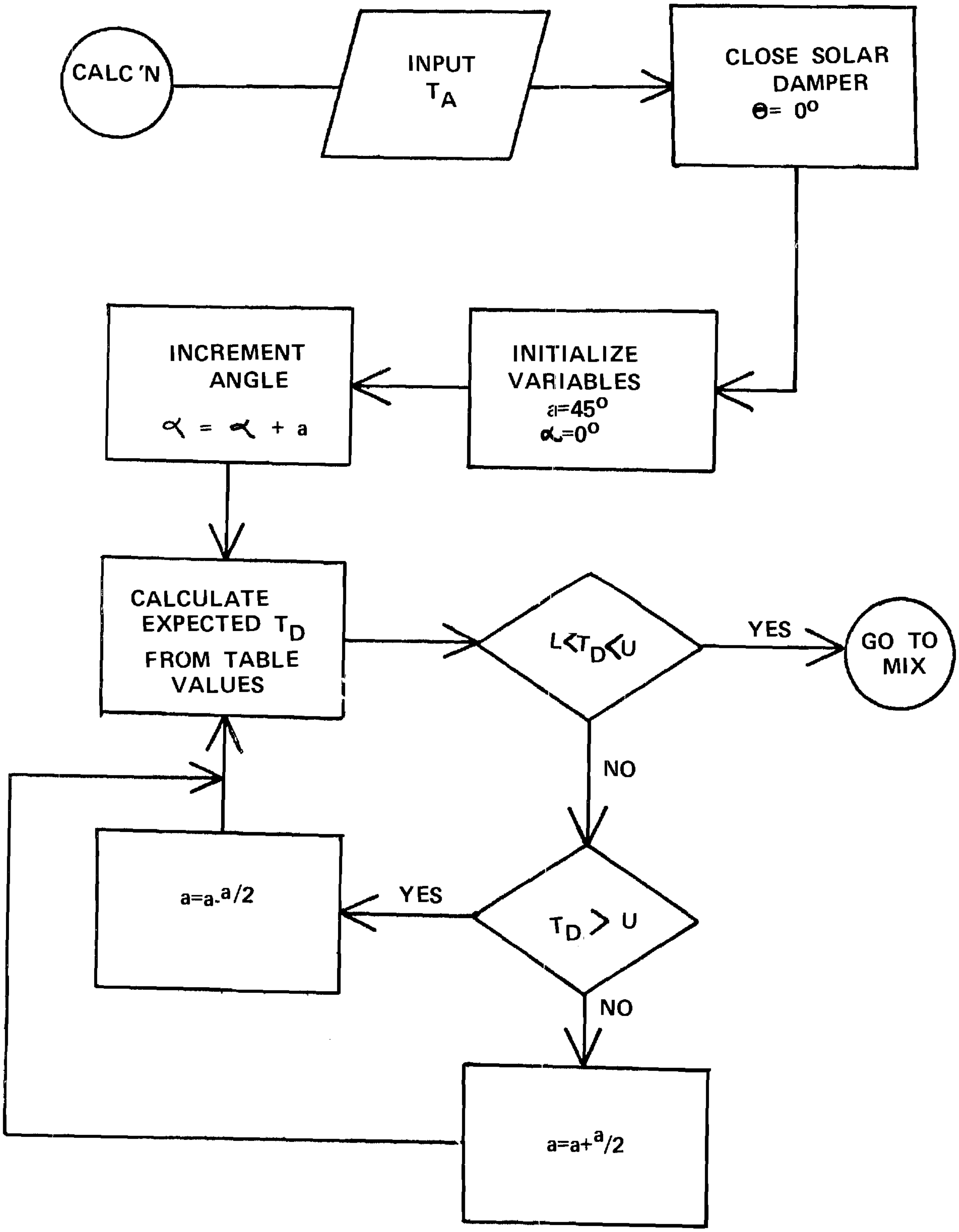
APPENDIX A

DETAILED FLOWCHARTS

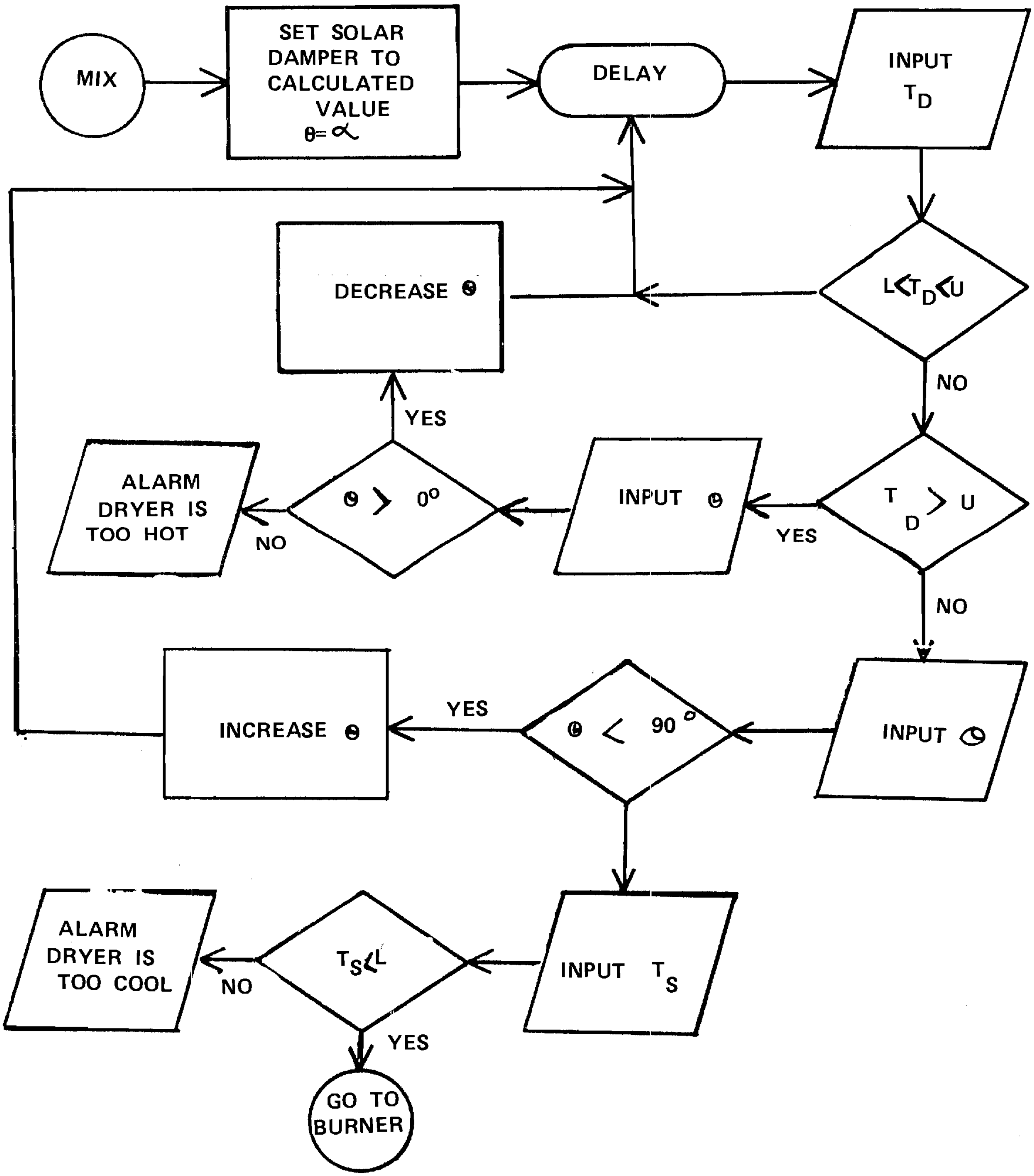
FOR CONTROLLING DAMPERS AND BURNERS

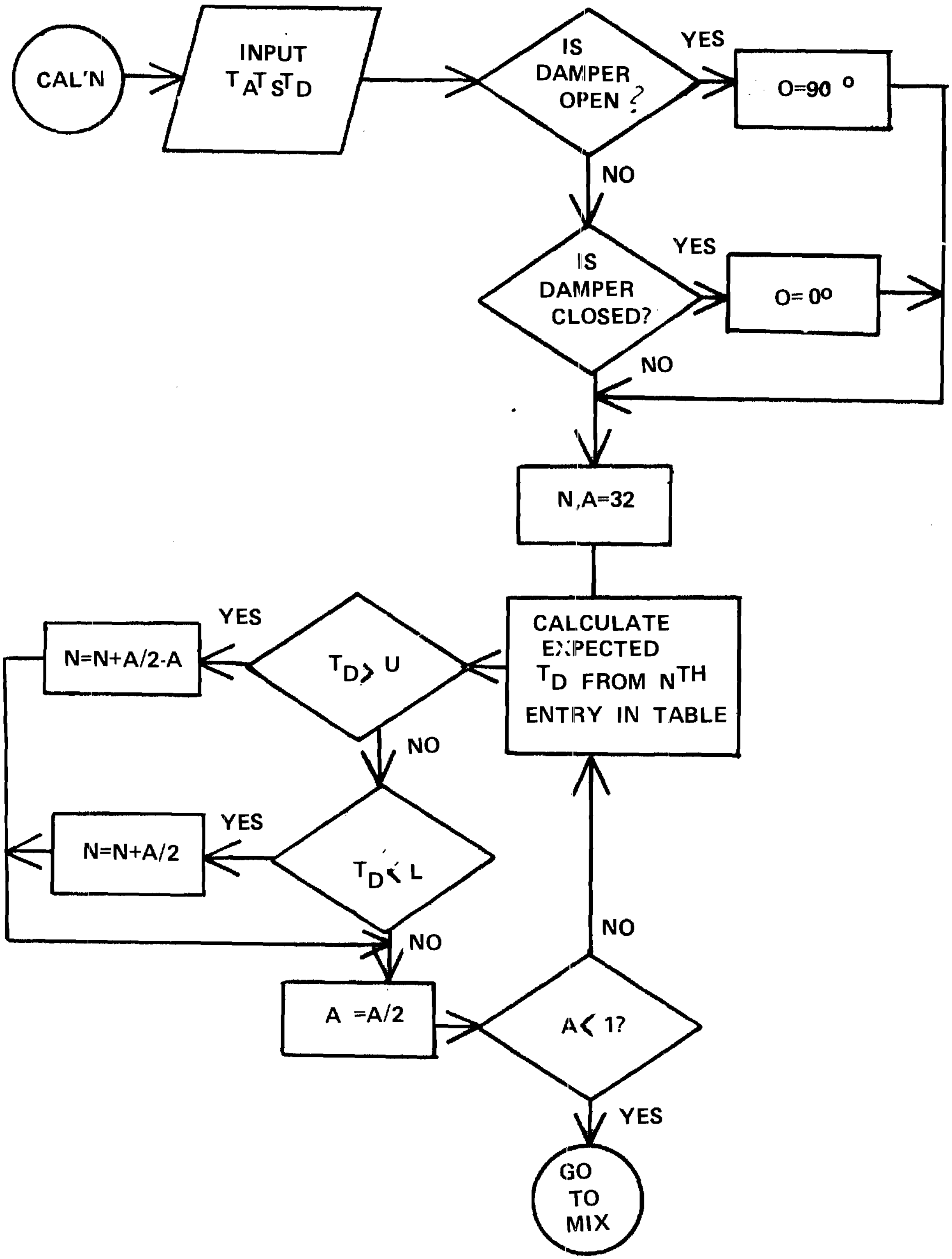

START
INPUT T_S
$L < T_S < U$
NO
YES
$T_S < L$
YES
GO TO BURNER
NO
GO TO CALC'N
OPEN SOLAR DAMPER TO MAX
MAX SOLAR:
$\Theta = 90^o$
DELAY
A


SET θ= 90°
TURN ON BURNER
DELAY
INPUT
T D
NO
T_D ≥ H
YES
H=L+ U-L / 2
BURNER
TURN OFF BURNER
INPUT
T_S
T_S<L
NO
GO TO A
YES
T_D<L
DELAY
INPUT
T_D



CALC 'N
INPUT T_A
CLOSE SOLAR DAMPER $\theta = 0^o$
INITIALIZE VARIABLES $a=45^o$ $\alpha=0^o$
INCREMENT ANGLE $\alpha = \alpha + a$
CALCULATE EXPECTED T_D FROM TABLE VALUES
$L < T_D < U$
YES
GO TO MIX
NO
$T_D > U$
YES
$a=a-a/2$
NO
$a=a+a/2$


MIX
SET SOLAR DAMPER TO CALCULATED VALUE θ=α
DELAY
INPUT T D
L<T D<U
NO
T D > U
YES
INPUT θ
θ > 0°
YES
DECREASE θ
NO
ALARM DRYER IS TOO HOT
NO
INPUT θ
θ < 90°
YES
INCREASE θ
INPUT T S
T S<L
NO
ALARM DRYER IS TOO COOL
YES
GO TO BURNER


CAL'N
INPUT
T_A T_S T_D
IS DAMPER OPEN ?
YES
O=90 °
NO
IS DAMPER CLOSED?
YES
O= 0°
NO
N,A=32
CALCULATE EXPECTED T_D FROM N^{TH} ENTRY IN TABLE
$T_D > U$
YES
N=N+A/2-A
NO
$T_D < L$
YES
N=N+A/2
NO
A =A/2
A< 1?
NO
YES
GO TO MIX

APPENDIX B

PROGRAM LISTING

PL/M-8Ø COMPILER

ISIS-II PL/M-8Ø V3.Ø COMPILATION OF MODULE USDA
OBJECT MODULE PLACED IN :F1:USDA.OBJ
COMPILER INVOKED BY: PLM8Ø :F1:USDA.SRC

```
  1          USDA:DO;          /*DATE:21 MAY 8Ø*/

                       /*SOLAR---LPN=1*/
                       /*BURNER--LPN=2*/
                       /*LOOP----LPN=3*/
                       /*MIX-----LPN=4*/
                       /*MAINT---LPN=5*/

  2   1          DECLARE OE4 BYTE INITIAL(ØFFH);
  3   1          DECLARE (I,U) ADDRESS DATA(3ØØ,32Ø);
  4   1          DECLARE (THETA,N,LPN,TS,TA,TD) ADDRESS;
  5   1          DECLARE (ALT,J,I,A) BYTE INITIAL(1,Ø,Ø,Ø);
  6   1          DECLARE KS(5Ø) ADDRESS DATA(3282,1689,1157,891,731,624,547,
                     489,444,408,378,353,331,313,296,282,269,258,247,238,
                     229,221,213,206,2ØØ,194,188,183,178,173,168,163,159,
                     155,151,143,14Ø,136,132,129,126,122,119,116,113,1Ø9,
                     1Ø6,1Ø3,1ØØ);
  7   1          DECLARE KA(5Ø) ADDRESS DATA(1ØØ,1Ø3,1Ø6,1Ø9,113,116,119,122,
                     126,129,132,136,14Ø,143,147,151,155,159,163,168,173,
                     178,183,188,194,2ØØ,2Ø6,213,221,229,238,247,258,269,
                     282,296,313,331,353,378,4Ø8,444,489,547,634,731,892,
                     1157,1689,3282);

  8   1          CO:PROCEDURE(CHAR) EXTERNAL;
  9   2             DECLARE CHAR BYTE;
 1Ø   2             END CO;

 11   1          NMOUT:PROCEDURE(ASCII) EXTERNAL;
 12   2                DECLARE ASCII BYTE;
 13   2                END NMOUT;


 14   1          TMP:PROCEDURE;
 15   2              CALL CO('L');
 16   2              CALL CO('P');
 17   2              CALL CO('N');
 18   2              CALL CO(' ');
 19   2              CALL NMOUT(LOW(LPN));

 2Ø   2              OE4=(ØF8H AND OE4) OR 5;
 21   2              OUTPUT(ØE4H)=OE4;             /*SELECT SOLAR*/
 22   2              OUTPUT(ØEAH)=1;
 23   2              OUTPUT(ØEAH)=Ø;      /*PULSE TO INIT CONV*/
 24   2              CALL CO(' ');
 25   2              CALL CO(' ');
 26   2              CALL CO('T');
 27   2              CALL CO('S');
 28   2              CALL CO('=');
 29   2              CALL TIME(1ØØ);
 3Ø   2              TS=(ØCØH AND INPUT(ØEAH))*4+(NOT INPUT(ØE8H));
 31   2              CALL NMOUT(HIGH(TS));
```

```
  32   2            CALL NMOUT(LOW(TS));


  33   2            OE4=OE4 OR 7;
  34   2            OUTPUT(0E4H)=OE4;             /*SELECT AMB TMP*/
  35   2            OUTPUT(0EAH)=1;
  36   2            OUTPUT(0EAH)=0;
  37   2            CALL CO(' ');
  38   2            CALL CO(' ');
  39   2            CALL CO('T');
  40   2            CALL CO('A');
  41   2            CALL CO('=');
  42   2            CALL TIME(100);
  43   2            TA=(0C0H AND INPUT(0EAH))*4+(NOT INPUT(0E8H));
  44   2            CALL NMOUT(HIGH(TA));
  45   2            CALL NMOUT(LOW(TA));

  46   2            OE4=(0F8H AND OE4) OR 6;          /*SELECT DRYER TMP*/
  47   2            OUTPUT(0E4H)=OE4;
  48   2            OUTPUT(0EAH)=1;
  49   2            OUTPUT(0EAH)=0;
  50   2            CALL CO(' ');
  51   2            CALL CO(' ');
  52   2            CALL CO('T');
  53   2            CALL CO('D');
  54   2            CALL CO('=');
  55   2            CALL TIME(100);
  56   2            TD=(0C0H AND INPUT(0EAH))*4+(NOT INPUT(0E8H));
  57   2            CALL NMOUT(HIGH(TD));
  58   2            CALL NMOUT(LOW(TD));

  59   2             CALL CO(' ');
  60   2             CALL CO(' ');
  61   2             CALL CO('T');
  62   2             CALL CO('H');
  63   2             CALL CO('E');
  64   2             CALL CO('T');
  65   2             CALL CO('A');
  66   2             CALL CO('=');
  67   2             CALL NMOUT(LOW(THETA));
  68   2             CALL CO(0DH);
  69   2             CALL CO(0AH);
  70   2        END TMP;

  71   1        FWD:PROCEDURE;  /*PULSES THE MOTOR CONTROL THAT
                      OPENS THE SOLAR DAMPER(1.8DEG/PULSE)*/

  72   2            OE4=OE4 AND 0EFH;    /*SET BIT 4--PORT E4
                               HAS INVERTED OUTPUTS*/
  73   2            OUTPUT(0E4H)=OE4;
  74   2            DO I=1 TO 5;
  75   3            CALL TIME(100);      /*10 MS DELAY*/
  76   3            END;
  77   2            OE4=OE4 OR 10H; /*ZERO BIT 4*/
  78   2            OUTPUT(0E4H)=OE4;
  79   2            DO I=1 TO 5;
  80   3            CALL TIME(100);
```

```
 81   3            END;
 82   2        END FWD;

 83   1        DELAY:PROCEDURE;
 84   2            DO I=1 TO 4;
 85   3            DO J=1 TO 25Ø;         /*1Ø SEC DELAY*/
 86   4            CALL TIME(1ØØ);
 87   4            END;
 88   3            END;
 89   2        END DELAY;

    /*THE FOLLOWING PROCEDURE OPENS SOLAR DAMPER TO MAX POSITION(9ØDEG

 90   1        DEG9Ø:PROCEDURE;
 91   2            DO WHILE(SHR(INPUT(ØE6H),1));
 92   3                CALL FWD;
 93   3            END;
 94   2            THETA=5Ø;
 95   2        END DEG9Ø;

 96   1        BKWD:PROCEDURE;       /*PULSES MOTOR CONTROL TO
                           CLOSE SOLAR DAMPER(1.8DEG/PULSE)*/

 97   2            OE4=OE4 AND ØF7H;    /*SET BIT 3*/
 98   2            OUTPUT(ØE4H)=OE4;    /*RAISE CONTROL LINE
                                 FOR BWK(CW) PULSES*/
 99   2            DO I=1 TO 5;
1ØØ   3            CALL TIME(1ØØ);      /*1Ø MS DELAY*/
1Ø1   3            END;
1Ø2   2            OE4=OE4 OR 8H;       /*ZERO BIT 3*/
1Ø3   2            OUTPUT(ØE4H)=OE4;    /*LOWER CONTROL LINE*/
1Ø4   2            DO I=1 TO 5;
1Ø5   3            CALL TIME(1ØØ);
1Ø6   3            END;
1Ø7   2        END BKWD;


1Ø8   1        INIT:OE4=ØFFH;             /*INITIALIZE CONTROL WORD
                                FOR PORT E4*/

1Ø9   1                OUTPUT(ØEBH)=98H;
11Ø   1             OUTPUT(ØE7H)=89H;

111   1        START:DO WHILE (INPUT(ØE6H));   /*CLOSE DAMPER TO SOLAR AIR*/
112   2              CALL BKWD;
113   2              END;
114   1               THETA=Ø;

115   1        SOLAR:LPN=1;
116   1            CALL CO(ØAH);
117   1            CALL CO(ØDH);
118   1            CALL CO('S');
119   1            CALL CO('O');
12Ø   1            CALL CO('L');
121   1            CALL CO('A');
122   1            CALL CO('R');
123   1            CALL CO(ØDH);
```

```
124   1            CALL CO(ØAH);
125   1            CALL TMP;
126   1            IF(TS>U)THEN GOTO MIX;
128   1            CALL DEG90;      /*TS<U,SOLAR AIR MAX*/
129   1            IF(TS<L)THEN GOTO BURNER;
131   1                ELSE CALL DELAY;    /*ELSE L<TS<U;SO STAY
                                IN SOLAR ROUTINE*/
132   1            GOTO SOLAR;

133   1        BURNER: LPN=2;
134   1            OUTPUT(ØE4H)=(OE4 AND ØDFH);    /*LIGHT BURNER*/
135   1        INTCON: CALL CO('B');
136   1            CALL CO('U');
137   1            CALL CO('R');
138   1            CALL CO('N');
139   1            CALL CO('E');
14Ø   1            CALL CO('R');
141   1            CALL CO(ØDH);
142   1            CALL CO(ØAH);
143   1            CALL TMP;

144   1            IF (TD<L+(U-L)/2) THEN CALL DELAY;
146   1            ELSE GO TO BRNOFF;
147   1            GO TO INTCON;
148   1        BRNOFF: OE4=OE4 OR 2ØH;      /*ZERO BIT 5 OF PORT1*/
149   1            OUTPUT(ØE4H)=OE4;   /*TURN OFF BURNER*/


15Ø   1        LOOP:LPN=3;
151   1            CALL CO('L');
152   1            CALL CO('O');
153   1            CALL CO('O');
154   1            CALL CO('P');
155   1            CALL CO(ØAH);
156   1            CALL CO(ØDH);
157   1            CALL TMP;
158   1            IF (TS<L)THEN CALL DELAY;
16Ø   1                    ELSE GOTO SOLAR;
161   1            CALL TMP;
162   1                IF(TD<L)THEN GOTO BURNER;  /*IF TD<L & TS<L*/
164   1                    ELSE GOTO LOOP; /*IF TD>L & TS<L*/

165   1        MIX:LPN=4;

166   1            CALL CO('M');
167   1            CALL CO('I');
168   1            CALL CO('X');
169   1            CALL CO(ØAH);
17Ø   1            CALL CO(ØDH);
171   1            CALL TMP;

              /*CHECK DAMPER POSITION; IF FULLY OPEN, THETA=5Ø*/

172   1            IF((INPUT(ØE6H) AND 1)=Ø)THEN THETA=Ø;
174   1            IF((INPUT(ØE6H) AND 2)=Ø)THEN THETA=5Ø;

                   /*ITERATIVE PROCEDURE FOR DETERMINING
```

```
                              THEORETICAL SETTING ,GIVEN TS & TA*/

 176    1                     N,A=32;
 177    1                     CALL CO('I');
 178    1                     CALL CO('T');
 179    1                     CALL CO('E');
 180    1                     CALL CO('R');
 181    1                     CALL CO(ØDH);
 182    1                     CALL CO(ØAH);
 183    1                ITER:   DO I=Ø TO 5;
 184    2                     TD=(TS*1Ø/KS(N)+TA*1Ø/KA(N))*1Ø;
 185    2                     A=SHR(A,1);
 186    2                     IF(TD>U)THEN N=N-SHL(A,1)+A;
 188    2                     IF(TD<L)THEN N=N+A;
 19Ø    2                     CALL CO(' ');
 191    2                     CALL CO('N');
 192    2                     CALL CO('=');
 193    2                     CALL NMOUT(LOW(N));
 194    2                     CALL CO(ØDH);
 195    2                     CALL CO(ØAH);
 196    2                     IF(N>49)THEN N=48;
 198    2                     END;
 199    1                                         DO WHILE (N<THETA);
 2ØØ    2                                         CALL BKWD;
 2Ø1    2                                         THETA=THETA-1;
 2Ø2    2                                         END;
 2Ø3    1                                         DO WHILE(N>THETA);
 2Ø4    2                                         CALL FWD;
 2Ø5    2                                         THETA=THETA+1;
 2Ø6    2                                         END;

 2Ø7    1                                    CALL CO('T');
 2Ø8    1                                    CALL CO('H');
 2Ø9    1                                    CALL CO('E');
 21Ø    1                                    CALL CO('T');
 211    1                                    CALL CO('A');
 212    1                                    CALL CO('=');
 213    1                                    CALL NMOUT(LOW(THETA));
 214    1                                    CALL CO(ØDH);
 215    1                                    CALL CO(ØAH);

                 /*THIS ROUTINE WILL MAINTAIN THE CORRECT DAMPER SETTING
                              FOR THE  DESIRED DRYER TEMPERATURE*/

 216    1                MAINT:  OE4=(OE4 AND ØF8H) OR 6;
 217    1                     OUTPUT(ØE4H)=OE4;            /*SELECT TD*/
 218    1                     OUTPUT(ØEAH)=1;
 219    1                     OUTPUT(ØEAH)=Ø;
 22Ø    1                     CALL CO('M');
 221    1                     CALL CO('A');
 222    1                     CALL CO('I');
 223    1                     CALL CO('N');
 224    1                     CALL CO('T');
 225    1                     CALL CO(ØAH);
 226    1                     CALL CO(ØDH);
 227    1                     CALL DELAY;      /*1Ø SEC DELAY*/
```

```
 228   1                 CALL TMP;

             /*THE FOLLOWING ROUTINES COMPENSATE FOR ANY POSSIBLE INACCURACIES
                 IN THE THEORETICAL CALCULATION FOR THE DAMPER SETTING*/

 229   1             DCR:    IF((TD>U) AND (THETA>Ø)) THEN DO;
 231   2                             THETA=THETA-1;
 232   2                             CALL FWD;
 233   2                             END;
 234   1             INCR:   IF((TD<L) AND (THETA<5Ø)) THEN DO;
 236   2                             CALL FWD;
 237   2                             THETA=THETA+1;
 238   2                             END;
 239   1                 CALL TMP;

             /*THIS ROUTINE WILL MAINTAIN DESIRED DRYER TEMP INDEFINITELY
                 UNLESS SOLAR TEMP DROPS BELOW LOWER LIMIT;
                 IN WHICH CASE, PROGRAM RETURNS TO SOLAR ROUTINE*/

 24Ø   1                 IF(TS<=L)THEN GOTO SOLAR;

             /*IF THE DRYER COMPARTMENT BECOMES TOO HOT OR TO COOL,
             AND THE DAMPER IS AT MIN OR MAX SETTING RESPECTIVELY,
             AN ALARM LIGHT WILL BE ACTIVATED AND THE PROGRAM HALTED*/

 242   1                 IF((TD>U) AND (THETA=Ø))THEN GO TO ALARM;
 244   1                 IF((TD<L) AND (THETA=5Ø))THEN GOTO ALARM;

 246   1                 GO TO MAINT;

 247   1             ALARM:  OE4=OE4 AND ØBFH;    /*SET ALARM BIT*/
 248   1                 OUTPUT(ØE4H)=OE4;    /*RAISE ALARM LINE*/
 249   1         END USDA;


MODULE INFORMATION:

     CODE AREA SIZE     = Ø6A4H   17ØØD
     VARIABLE AREA SIZE = ØØ11H     17D
     MAXIMUM STACK SIZE = ØØØ6H      6D
     312 LINES READ
     Ø PROGRAM ERROR(S)

END OF PL/M-80 COMPILATION
```

```
ISIS-II PL/M-80 V3.0 COMPILATION OF MODULE USDA
OBJECT MODULE PLACED IN :F1:USDA.OBJ
COMPILER INVOKED BY:  PLM80 :F1:USDA.SRC CODE



   1            USDA:DO;          /*DATE:21 MAY 80*/

                      /*SOLAR---LPN=1*/
                      /*BURNER--LPN=2*/
                      /*LOOP----LPN=3*/
                      /*MIX-----LPN=4*/
                      /*MAINT---LPN=5*/

   2    1           DECLARE OE4 BYTE INITIAL(0FFH);
   3    1           DECLARE (L,U) ADDRESS DATA(300,320);
   4    1           DECLARE (THETA,N,LPN,TS,TA,TD) ADDRESS;
   5    1           DECLARE (ALT,J,I,A) BYTE INITIAL(1,0,0,0);
   6    1           DECLARE KS(50) ADDRESS DATA(3282,1689,1157,891,731,624,547,
                        489,444,408,378,353,331,313,296,282,269,258,247,238,
                        229,221,213,206,200,194,188,183,178,173,168,163,159,
                        155,151,143,140,136,132,129,126,122,119,116,113,109,
                        106,103,100);
   7    1           DECLARE KA(50) ADDRESS DATA(100,103,106,109,113,116,119,122,
                        126,129,132,136,140,143,147,151,155,159,163,168,173,
                        178,183,188,194,200,206,213,221,229,238,247,258,269,
                        282,296,313,331,353,378,408,444,489,547,634,731,892,
                        1157,1689,3282);

   8    1           CO:PROCEDURE(CHAR) EXTERNAL;
   9    2              DECLARE CHAR BYTE;
  10    2              END CO;

  11    1           NMOUT:PROCEDURE(ASCII) EXTERNAL;
  12    2                 DECLARE ASCII BYTE;
  13    2                 END NMOUT;


  14    1           TMP:PROCEDURE;
                                                        ; STATEMENT # 14

                          ; PROC  TMP
  15    2                CALL CO('L');
                                                        ; STATEMENT # 15
            0479  0E4C            MVI     C,4CH
            047B  CD0000          CALL    CO
  16    2                CALL CO('P');
                                                        ; STATEMENT # 16
            047E  0E50            MVI     C,50H
            0480  CD0000          CALL    CO
  17    2                CALL CO('N');
                                                        ; STATEMENT # 17
            0483  0E4E            MVI     C,4EH
            0485  CD0000          CALL    CO
  18    2                CALL CO(' ');
                                                        ; STATEMENT # 18
            0488  0E20            MVI     C,20H
            048A  CD0000          CALL    CO
```

```
 19   2              CALL NMOUT(LOW(LPN));
                                                        ; STATEMENT # 19
          048D  2A0500            LHLD     LPN
          0490  7D                MOV      A,L
          0491  4F                MOV      C,A
          0492  CD0000            CALL     NMOUT

 20   2              OE4=(0F8H AND OE4) OR 5;
                                                        ; STATEMENT # 20
          0495  3A0000            LDA      OE4
          0498  E6F8              ANI      0F8H
          049A  F605              ORI      5H
          049C  320000            STA      OE4
 21   2              OUTPUT(0E4H)=OE4;                   /*SELECT SOLAR*/
                                                        ; STATEMENT # 21
          049F  D3E4              OUT      0E4H
 22   2              OUTPUT(0EAH)=1;
                                                        ; STATEMENT # 22
          04A1  3E01              MVI      A,1H
          04A3  D3EA              OUT      0EAH
 23   2              OUTPUT(0EAH)=0;        /*PULSE TO INIT CONV*/
                                                        ; STATEMENT # 23
          04A5  3E00              MVI      A,0H
          04A7  D3EA              OUT      0EAH
 24   2              CALL CO(' ');
                                                        ; STATEMENT # 24
          04A9  0E20              MVI      C,20H
          04AB  CD0000            CALL     CO
 25   2              CALL CO(' ');
                                                        ; STATEMENT # 25
          04AE  0E20              MVI      C,20H
          04B0  CD0000            CALL     CO
 26   2              CALL CO('T');
                                                        ; STATEMENT # 26
          04B3  0E54              MVI      C,54H
          04B5  CD0000            CALL     CO
 27   2              CALL CO('S');
                                                        ; STATEMENT # 27
          04B8  0E53              MVI      C,53H
          04BA  CD0000            CALL     CO
 28   2              CALL CO('=');
                                                        ; STATEMENT # 28
          04BD  0E3D              MVI      C,3DH
          04BF  CD0000            CALL     CO
 29   2              CALL TIME(100);
                                                        ; STATEMENT # 29
          04C2  3E64              MVI      A,64H
          04C4  CD0000            CALL     @P0105
 30   2              TS=(0C0H AND INPUT(0EAH))*4+(NOT INPUT(0E8H));
                                                        ; STATEMENT # 30
          04C7  DBEA              IN       0EAH
          04C9  E6C0              ANI      0C0H
          04CB  6F                MOV      L,A
          04CC  2600              MVI      H,0
          04CE  29                DAD      H
          04CF  29                DAD      H
          04D0  DBE8              IN       0E8H
```

```
             04D2   2F                  CMA
             04D3   5F                  MOV       E,A
             04D4   1600                MVI       D,0
             04D6   19                  DAD       D
             04D7   220700              SHLD      TS
  31    2                  CALL NMOUT(HIGH(TS));
                                                           ; STATEMENT # 31
             04DA   2A0700              LHLD      TS
             04DD   7C                  MOV       A,H
             04DE   4F                  MOV       C,A
             04DF   CD0000              CALL      NMOUT
  32    2                  CALL NMOUT(LOW(TS));
                                                           ; STATEMENT # 32
             04E2   2A0700              LHLD      TS
             04E5   7D                  MOV       A,L
             04E6   4F                  MOV       C,A
             04E7   CD0000              CALL      NMOUT


  33    2                  OE4=OE4 OR 7;
                                                           ; STATEMENT # 33
             04EA   3A0000              LDA       OE4
             04ED   F607                ORI       7H
             04EF   320000              STA       OE4
  34    2                  OUTPUT(0E4H)=OE4;                /*SELECT AMB TMP*/
                                                           ; STATEMENT # 34
             04F2   D3E4                OUT       0E4H
  35    2                  OUTPUT(0EAH)=1;
                                                           ; STATEMENT # 35
             04F4   3E01                MVI       A,1H
             04F6   D3EA                OUT       0EAH
  36    2                  OUTPUT(0EAH)=0;
                                                           ; STATEMENT # 36
             04F8   3E00                MVI       A,0H
             04FA   D3EA                OUT       0EAH
  37    2                  CALL CO(' ');
                                                           ; STATEMENT # 37
             04FC   0E20                MVI       C,20H
             04FE   CD0000              CALL      CO
  38    2                  CALL CO(' ');
                                                           ; STATEMENT # 38
             0501   0E20                MVI       C,20H
             0503   CD0000              CALL      CO
  39    2                  CALL CO('T');
                                                           ; STATEMENT # 39
             0506   0E54                MVI       C,54H
             0508   CD0000              CALL      CO
  40    2                  CALL CO('A');
                                                           ; STATEMENT # 40
             050B   0E41                MVI       C,41H
             050D   CD0000              CALL      CO
  41    2                  CALL CO('=');
                                                           ; STATEMENT # 41
             0510   0E3D                MVI       C,3DH
             0512   CD0000              CALL      CO
  42    2                  CALL TIME(100);
                                                           ; STATEMENT # 42
```

```
          Ø515   3E64             MVI      A,64H
          Ø517   CDØØØØ           CALL     @PØ1Ø5
 43   2                 TA=(ØCØH AND INPUT(ØEAH))*4+(NOT INPUT(ØF8H));
                                                    ; STATEMENT # 43
          Ø51A   DBEA             IN       ØEAH
          Ø51C   E6CØ             ANI      ØCØH
          Ø51E   6F               MOV      L,A
          Ø51F   26ØØ             MVI      H,Ø
          Ø521   29               DAD      H
          Ø522   29               DAD      H
          Ø523   DBF8             IN       ØF8H
          Ø525   2F               CMA
          Ø526   5F               MOV      E,A
          Ø527   16ØØ             MVI      D,Ø
          Ø529   19               DAD      D
          Ø52A   22Ø9ØØ           SHLD     TA
 44   2                 CALL NMOUT(HIGH(TA));
                                                    ; STATEMENT # 44
          Ø52D   2AØ9ØØ           LHLD     TA
          Ø53Ø   7C               MOV      A,H
          Ø531   4F               MOV      C,A
          Ø532   CDØØØØ           CALL     NMOUT
 45   2                 CALL NMOUT(LOW(TA));
                                                    ; STATEMENT # 45
          Ø535   2AØ9ØØ           LHLD     TA
          Ø538   7D               MOV      A,L
          Ø539   4F               MOV      C,A
          Ø53A   CDØØØØ           CALL     NMOUT

 46   2                 OE4=(ØF8H AND OE4) OR 6;        /*SELECT DRYER TMP*/
                                                    ; STATEMENT # 46
          Ø53D   3AØØØØ           LDA      OE4
          Ø54Ø   E6F8             ANI      ØF8H
          Ø542   F6Ø6             ORI      6H
          Ø544   32ØØØØ           STA      OE4
 47   2                 OUTPUT(ØE4H)=OE4;
                                                    ; STATEMENT # 47
          Ø547   D3E4             OUT      ØE4H
 48   2                 OUTPUT(ØEAH)=1;
                                                    ; STATEMENT # 48
          Ø549   3EØ1             MVI      A,1H
          Ø54B   D3EA             OUT      ØEAH
 49   2                 OUTPUT(ØEAH)=Ø;
                                                    ; STATEMENT # 49
          Ø54D   3EØØ             MVI      A,ØH
          Ø54F   D3EA             OUT      ØEAH
 5Ø   2                 CALL CO(' ');
                                                    ; STATEMENT # 5Ø
          Ø551   ØE2Ø             MVI      C,2ØH
          Ø553   CDØØØØ           CALL     CO
 51   2                 CALL CO(' ');
                                                    ; STATEMENT # 51
          Ø556   ØE2Ø             MVI      C,2ØH
          Ø558   CDØØØØ           CALL     CO
 52   2                 CALL CO('T');
                                                    ; STATEMENT # 52
          Ø55B   ØE54             MVI      C,54H
```

```
            055D  CD0000            CALL    CO
  53   2                CALL CO('D');
                                                       ; STATEMENT # 53
            0560  0E44              MVI     C,44H
            0562  CD0000            CALL    CO
  54   2                CALL CO('=');
                                                       ; STATEMENT # 54
            0565  0E3D              MVI     C,3DH
            0567  CD0000            CALL    CO
  55   2                CALL TIME(100);
                                                       ; STATEMENT # 55
            056A  3E64              MVI     A,64H
            056C  CD0000            CALL    @P0105
  56   2                TD=(0C0H AND INPUT(0EAH))*4+(NOT INPUT(0E8H));
                                                       ; STATEMENT # 56
            056F  DBEA              IN      0EAH
            0571  E6C0              ANI     0C0H
            0573  6F                MOV     L,A
            0574  2600              MVI     H,0
            0576  29                DAD     H
            0577  29                DAD     H
            0578  DBE8              IN      0E8H
            057A  2F                CMA
            057B  5F                MOV     E,A
            057C  1600              MVI     D,0
            057E  19                DAD     D
            057F  220B00            SHLD    TD
  57   2                CALL NMOUT(HIGH(TD));
                                                       ; STATEMENT # 57
            0582  2A0B00            LHLD    TD
            0585  7C                MOV     A,H
            0586  4F                MOV     C,A
            0587  CD0000            CALL    NMOUT
  58   2                CALL NMOUT(LOW(TD));
                                                       ; STATEMENT # 58
            058A  2A0B00            LHLD    TD
            058D  7D                MOV     A,L
            058E  4F                MOV     C,A
            058F  CD0000            CALL    NMOUT

  59   2                 CALL CO(' ');
                                                       ; STATEMENT # 59
            0592  0E20              MVI     C,20H
            0594  CD0000            CALL    CO
  60   2                 CALL CO(' ');
                                                       ; STATEMENT # 60
            0597  0E20              MVI     C,20H
            0599  CD0000            CALL    CO
  61   2                 CALL CO('T');
                                                       ; STATEMENT # 61
            059C  0E54              MVI     C,54H
            059E  CD0000            CALL    CO
  62   2                 CALL CO('H');
                                                       ; STATEMENT # 62
            05A1  0E48              MVI     C,48H
            05A3  CD0000            CALL    CO
  63   2                 CALL CO('E');
```

```
                                                   ; STATEMENT # 63
          05A6   0E45             MVI      C,45H
          05A8   CD0000           CALL     CO
 64   2               CALL CO('T');
                                                   ; STATEMENT # 64
          05AB   0E54             MVI      C,54H
          05AD   CD0000           CALL     CO
 65   2               CALL CO('A');
                                                   ; STATEMENT # 65
          05B0   0E41             MVI      C,41H
          05B2   CD0000           CALL     CO
 66   2               CALL CO('=');
                                                   ; STATEMENT # 66
          05B5   0E3D             MVI      C,3DH
          05B7   CD0000           CALL     CO
 67   2               CALL NMOUT(LOW(THETA));
                                                   ; STATEMENT # 67
          05BA   2A0100           LHLD     THETA
          05BD   7D               MOV      A,L
          05BE   4F               MOV      C,A
          05BF   CD0000           CALL     NMOUT
 68   2               CALL CC(0DH);
                                                   ; STATEMENT # 68
          05C2   0E0D             MVI      C,0DH
          05C4   CD0000           CALL     CO
 69   2               CAIL CO(0AH);
                                                   ; STATEMENT # 69
          05C7   0E0A             MVI      C,0AH
          05C9   CD0000           CALL     CO
 70   2          END TMP;
                                                   ; STATEMENT # 70
          05CC   C9               RET

 71   1          FWD:PROCEDURE;  /*PULSES THE MOTOR CONTROL THAT
                                                   ; STATEMENT # 71
                        ; PROC  FWD
                         OPENS THE SOLAR DAMPER(1.8DEG/PULSE)*/

 72   2              OE4=OE4 AND 0EFH;   /*SET BIT 4--PORT E4
                                                   ; STATEMENT # 72
                                  HAS INVERTED OUTPUTS*/
          05CD   3A0000           LDA      OE4
          05D0   E6EF             ANI      0EFH
          05D2   320000           STA      OE4
 73   2              OUTPUT(0E4H)=OE4;
                                                   ; STATEMENT # 73
          05D5   D3E4             OUT      0E4H
 74   2              DO I=1 TO 5;
                                                   ; STATEMENT # 74
          05D7   210F00           LXI      H,I
          05DA   3601             MVI      M,1H
                        @20:
          05DC   3E05             MVI      A,5H
          05DE   210F00           LXI      H,I
          05E1   BE               CMP      M
          05E2   DAF105           JC       @21
 75   3              CAIL TIME(100);     /*10 MS DELAY*/
```

```
                                                        ; STATEMENT # 75
            05E5  3E64               MVI      A,64H
            05E7  CD0000             CALL     @P0105
  76   3                END;
                                                        ; STATEMENT # 76
            05EA  210F00             LXI      H,I
            05ED  34                 INR      M
            05EF  C2DC05             JNZ      @20
                           @21:
  77   2                OE4=OE4 OR 10H; /*ZERO BIT 4*/
                                                        ; STATEMENT # 77
            05F1  3A0000             LDA      OE4
            05F4  F610               ORI      10H
            05F6  320000             STA      OE4
  78   2                OUTPUT(0E4H)=OE4;
                                                        ; STATEMENT # 78
            05F9  D3E4               OUT      0E4H
  79   2                DO I=1 TO 5;
                                                        ; STATEMENT # 79
            05FB  210F00             LXI      H,I
            05FE  3601               MVI      M,1H
                           @22:
            0600  3E05               MVI      A,5H
            0602  210F00             LXI      H,I
            0605  BE                 CMP      M
            0606  DA1506             JC       @23
  80   3                CAII TIME(100);
                                                        ; STATEMENT # 80
            0609  3E64               MVI      A,64H
            060B  CD0000             CALL     @P0105
  81   3                END;
                                                        ; STATEMENT # 81
            060E  210F00             LXI      H,I
            0611  34                 INR      M
            0612  C20006             JNZ      @22
                           @23:
  82   2            END FWD;
                                                        ; STATEMENT # 82
            0615  C9                 RET

  83   1            DELAY:PROCEDURE;
                                                        ; STATEMENT # 83
                           ; PROC  DELAY
  84   2                DO I=1 TO 4;
                                                        ; STATEMENT # 84
            0616  210F00             LXI      H,I
            0619  3601               MVI      M,1H
                           @24:
            061B  3E04               MVI      A,4H
            061D  210F00             LXI      H,I
            0620  BE                 CMP      M
            0621  DA4506             JC       @25
  85   3                DO J=1 TO 250;           /*10 SEC DELAY*/
                                                        ; STATEMENT # 85
            0624  210E00             LXI      H,J
            0627  3601               MVI      M,1H
                           @26:
```

```
         0629  3EFA              MVI     A,0FAH
         062B  210E00            LXI     H,J
         062E  BE                CMP     M
         062F  DA3E06            JC      @27
 86   4              CALL TIME(100);
                                                 ; STATEMENT # 86
         0632  3E64              MVI     A,64H
         0634  CD0000            CALL    @P0105
 87   4              END;
                                                 ; STATEMENT # 87
         0637  210E00            LXI     H,J
         063A  34                INR     M
         063B  C22906            JNZ     @26
                        @27:
 88   3              END;
                                                 ; STATEMENT # 88
         063E  210F00            LXI     H,I
         0641  34                INR     M
         0642  C21B06            JNZ     @24
                        @25:
 89   2          END DELAY;
                                                 ; STATEMENT # 89
         0645  C9                RET

      /*THE FOLLOWING PROCEDURE OPENS SOLAR DAMPER TO MAX POSITION(90DEG)

 90   1          DEG90:PROCEDURE;
                                                 ; STATEMENT # 90
                        ; PROC  DEG90
 91   2              DO WHILE(SHR(INPUT(0E6H),1));
                                                 ; STATEMENT # 91
                        @28:
         0646  DBE6              IN      0E6H
         0648  B7                ORA     A
         0649  1F                RAR
         064A  1F                RAR
         064B  D25406            JNC     @29
 92   3                  CALL FWD;
                                                 ; STATEMENT # 92
         064E  CDCD05            CALL    FWD
 93   3              END;
                                                 ; STATEMENT # 93
         0651  C34606            JMP     @28
                        @29:
 94   2              THETA=50;
                                                 ; STATEMENT # 94
         0654  213200            LXI     H,32H
         0657  220100            SHLD    THETA
 95   2          END DEG90;
                                                 ; STATEMENT # 95
         065A  C9                RET

 96   1          BKWD:PROCEDURE;     /*PULSES MOTOR CONTROL TO
                                                 ; STATEMENT # 96
                        ; PROC  BKWD
                             CLOSE SOLAR DAMPER(1.8DEG/PULSE)*/
```

```
 97   2               OE4=OE4 AND 0F7H;    /*SET BIT 3*/
                                                 ; STATEMENT # 97
          065B  3A0000           LDA      OE4
          065E  E6F7             ANI      0F7H
          0660  320000           STA      OE4
 98   2               OUTPUT(0E4H)=OE4;    /*RAISE CONTROL LINE
                                                 ; STATEMENT # 98
                                  FOR BWK(CW) PULSES*/
          0663  D3E4             OUT      0E4H
 99   2               DO I=1 TO 5;
                                                 ; STATEMENT # 99
          0665  210F00           LXI      H,I
          0668  3601             MVI      M,1H
                          @30:
          066A  3E05             MVI      A,5H
          066C  210F00           LXI      H,I
          066F  BE               CMP      M
          0670  DA7F06           JC       @31
100   3               CALL TIME(100);       /*10 MS DELAY*/
                                                 ; STATEMENT # 100
          0673  3E64             MVI      A,64H
          0675  CD0000           CALL     @P0105
101   3               END;
                                                 ; STATEMENT # 101
          0678  210F00           LXI      H,I
          067B  34               INR      M
          067C  C26A06           JNZ      @30
                          @31:
102   2               OE4=OE4 OR 8H;        /*ZERO BIT 3*/
                                                 ; STATEMENT # 102
          067F  3A0000           LDA      OE4
          0682  F608             ORI      8H
          0684  320000           STA      OE4
103   2               OUTPUT(0E4H)=OE4;     /*LOWER CONTROL LINE*/
                                                 ; STATEMENT # 103
          0687  D3E4             OUT      0E4H
104   2               DO I=1 TO 5;
                                                 ; STATEMENT # 104
          0689  210F00           LXI      H,I
          068C  3601             MVI      M,1H
                          @32:
          068E  3E05             MVI      A,5H
          0690  210F00           LXI      H,I
          0693  BE               CMP      M
          0694  DAA306           JC       @33
105   3               CALL TIME(100);
                                                 ; STATEMENT # 105
          0697  3E64             MVI      A,64H
          0699  CD0000           CALL     @P0105
106   3               END;
                                                 ; STATEMENT # 106
          069C  210F00           LXI      H,I
          069F  34               INR      M
          06A0  C28E06           JNZ      @32
                          @33:
107   2            END BKWD;
                                                 ; STATEMENT # 107
```

```
                36A3   C9                    RET


 108    1              INIT:OE4=ØFFH;             /*INITIALIZE CONTROL WORD
                                                         ; STATEMENT # 108
                ØØCC   31ØØØØ                LXI      SP,@STACK$ORIGIN
                                      INIT:
                                                  FOR PORT E4*/

                ØØCF   21ØØØØ                LXI      H,OE4
                ØØD2   36FF                  MVI      M,ØFFH
 109    1                        OUTPUT(ØEBH)=98H;
                                                         ; STATEMENT # 109
                ØØD4   3E98                  MVI      A,98H
                ØØD6   D3EB                  OUT      ØEBH
 11Ø    1                   OUTPUT(ØE7H)=89H;
                                                         ; STATEMENT # 11Ø

                ØØD8   3E89                  MVI      A,89H
                ØØDA   D3E7                  OUT      ØE7H
 111    1              START:DO WHILE (INPUT(ØE6H));    /*CLOSE DAMPER TO SOLAR AIR*/
                                                         ; STATEMENT # 111
                                      START:
                                      @34:
                ØØDC   DBE6                  IN       ØE6H
                ØØDE   1F                    RAR
                ØØDF   D2E8ØØ                JNC      @35
 112    2                    CALL BKWD;
                                                         ; STATEMENT # 112
                ØØE2   CD5BØ6                CALL     BKWD
 113    2                    END;
                                                         ; STATEMENT # 113
                ØØE5   C3DCØØ                JMP      @34
                                      @35:
 114    1                             THETA=Ø;
                                                         ; STATEMENT # 114

                ØØE8   21ØØØØ                LXI      H,ØH
                ØØEB   22Ø1ØØ                SHLD     THETA
 115    1              SOLAR:LPN=1;
                                                         ; STATEMENT # 115
                                      SOLAR:
                ØØEE   21Ø1ØØ                LXI      H,1H
                ØØF1   22Ø5ØØ                SHLD     LPN
 116    1                   CALL CO(ØAH);
                                                         ; STATEMENT # 116
                ØØF4   ØEØA                  MVI      C,ØAH
                ØØF6   CDØØØØ                CALL     CO
 117    1                   CALL CO(ØDH);
                                                         ; STATEMENT # 117
                ØØF9   ØEØD                  MVI      C,ØDH
                ØØFB   CDØØØØ                CALL     CO
 118    1                   CALL CO('S');
                                                         ; STATEMENT # 118
                ØØFE   ØE53                  MVI      C,53H
                Ø1ØØ   CDØØØØ                CALL     CO
 119    1                   CALL CO('O');
```

```
                                                    ; STATEMENT # 119
          0103  0E4F              MVI      C,4FH
          0105  CD0000            CALL     CO
 120   1               CALL CO('I');
                                                    ; STATEMENT # 120
          0108  0E4C              MVI      C,4CH
          010A  CD0000            CALL     CO
 121   1               CALL CO('A');
                                                    ; STATEMENT # 121
          010D  0E41              MVI      C,41H
          010F  CD0000            CALL     CO
 122   1               CALL CO('R');
                                                    ; STATEMENT # 122
          0112  0E52              MVI      C,52H
          0114  CD0000            CALL     CO
 123   1               CALL CO(0DH);
                                                    ; STATEMENT # 123
          0117  0E0D              MVI      C,0DH
          0119  CD0000            CALL     CO
 124   1               CALL CO(0AH);
                                                    ; STATEMENT # 124
          011C  0E0A              MVI      C,0AH
          011E  CD0000            CALL     CO
 125   1               CALL TMP;
                                                    ; STATEMENT # 125
          0121  CD7904            CALL     TMP
 126   1               IF(TS>U)THEN GOTO MIX;
                                                    ; STATEMENT # 126
          0124  110200            LXI      D,U
          0127  010700            LXI      B,TS
          012A  CD0000            CALL     @P0098
          012D  D23301            JNC      @1
                                                    ; STATEMENT # 127
          0130  C30702            JMP      MIX
                          @1:
 128   1               CALL DEG90;      /*TS<U,SOLAR AIR MAX*/
                                                    ; STATEMENT # 128
          0133  CD4606            CALL     DEG90
 129   1               IF(TS<L)THEN GOTO BURNER;
                                                    ; STATEMENT # 129
          0136  010000            LXI      B,L
          0139  110700            LXI      D,TS
          013C  CD0000            CALL     @P0098
          013F  D24501            JNC      @2
                                                    ; STATEMENT # 130
          0142  C34B01            JMP      BURNER
                          @2:
 131   1                  ELSE CALL DELAY;    /*ELSE L<TS<U;SO STAY
                                                    ; STATEMENT # 131
          0145  CD1606            CALL     DELAY
                          @3:
                                          IN SOLAR ROUTINE*/
 132   1               GOTO SOLAR;
                                                    ; STATEMENT # 132
          0148  C3FE00            JMP      SOLAR

 133   1           BURNER: LPN=2;
```

```
                                                          ; STATEMENT # 133
                          BURNER:
           014B  210200            LXI     H,2H
           014E  220500            SHLD    LPN
 134   1                CUTPUT(0E4H)=(OE4 AND 0DFH);    /*LIGHT BURNER*/
                                                          ; STATEMENT # 134
           0151  3A0000            LDA     OE4
           0154  E6DF              ANI     0DFH
           0156  D3E4              OUT     0E4H
 135   1             INTCON: CALL CO('B');
                                                          ; STATEMENT # 135
                          INTCON:
           0158  0E42              MVI     C,42H
           015A  CD0000            CALL    CO
 136   1                CALL CO('U');
                                                          ; STATEMENT # 136
           015D  0E55              MVI     C,55H
           015F  CD0000            CALL    CO
 137   1                CALL CO('R');
                                                          ; STATEMENT # 137
           0162  0E52              MVI     C,52H
           0164  CD0000            CALL    CO
 138   1                CALL CO('N');
                                                          ; STATEMENT # 138
           0167  0E4E              MVI     C,4EH
           0169  CD0000            CALL    CO
 139   1                CALL CO('E');
                                                          ; STATEMENT # 139
           016C  0E45              MVI     C,45H
           016E  CD0000            CALL    CO
 140   1                CALL CO('R');
                                                          ; STATEMENT # 140
           0171  0E52              MVI     C,52H
           0173  CD0000            CALL    CO
 141   1                CALL CO(0DH);
                                                          ; STATEMENT # 141
           0176  0E0D              MVI     C,0DH
           0178  CD0000            CALL    CO
 142   1                CALL CO(0AH);
                                                          ; STATEMENT # 142
           017B  0E0A              MVI     C,0AH
           017D  CD0000            CALL    CO
 143   1                CALL TMP;
                                                          ; STATEMENT # 143

           0180  CD7904            CALL    TMP
 144   1                IF (TD<L+(U-L)/2) THEN CALL DELAY;
                                                          ; STATEMENT # 144
           0183  010000            LXI     B,L
           0186  110200            LXI     D,U
           0189  CD0000            CALL    @P0098
           018C  EB                XCHG
           018D  210200            LXI     H,2H
           0190  CD0000            CALL    @P0029
           0193  2A0000            LHLD    L
           0196  19                DAD     D
           0197  110B00            LXI     D,TD
```

```
                019A   CD0000          CALL    @P0102
                019D   D2A601          JNC     @4
                                                         ; STATEMENT # 145
                01A0   CD1606          CALL    DELAY
                01A3   C3A901          JMP     @5
                               @4:
 146    1                   ELSE GO TO BRNOFF;
                                                         ; STATEMENT # 146
                01A6   C3AC01          JMP     BRNOFF
                               @5:
 147    1                   GO TO INTCON;
                                                         ; STATEMENT # 147
                01A9   C35801          JMP     INTCON
 148    1               BRNOFF: OE4=OE4 OR 20H;     /*ZERO BIT 5 OF PORT1*/
                                                         ; STATEMENT # 148
                               BRNOFF:
                01AC   3A0000          LDA     OE4
                01AF   F620            ORI     20H
                01B1   320000          STA     OE4
 149    1                   OUTPUT(0E4H)=OE4;    /*TURN OFF BURNER*/
                                                         ; STATEMENT # 149


                01B4   D3E4            OUT     0E4H
 150    1               LOOP:LPN=3;
                                                         ; STATEMENT # 150
                               LOOP:
                01B6   210300          LXI     H,3H
                01B9   220500          SHLD    LPN
 151    1                   CALL CO('L');
                                                         ; STATEMENT # 151
                01BC   0E4C            MVI     C,4CH
                01BE   CD0000          CALL    CO
 152    1                   CALL CO('O');
                                                         ; STATEMENT # 152
                01C1   0E4F            MVI     C,4FH
                01C3   CD0000          CALL    CO
 153    1                   CALL CO('O');
                                                         ; STATEMENT # 153
                01C6   0E4F            MVI     C,4FH
                01C8   CD0000          CALL    CO
 154    1                   CALL CO('P');
                                                         ; STATEMENT # 154
                01CB   0E50            MVI     C,50H
                01CD   CD0000          CALL    CO
 155    1                   CALL CO(0AH);
                                                         ; STATEMENT # 155
                01D0   0E0A            MVI     C,0AH
                01D2   CD0000          CALL    CO
 156    1                   CALL CO(0DH);
                                                         ; STATEMENT # 156
                01D5   0E0D            MVI     C,0DH
                01D7   CD0000          CALL    CO
 157    1                   CALL TMP;
                                                         ; STATEMENT # 157
                01DA   CD7904          CALL    TMP
 158    1                   IF (TS<L)THEN CALL DELAY;
```

```
                                                        ; STATEMENT # 158
           01DD  010000          LXI     B,L
           01E0  112720          LXI     D,TS
           01E3  CD0000          CALL    @P0098
           01E6  D2EF01          JNC     @6
                                                        ; STATEMENT # 159
           01E9  CD1606          CALL    DELAY
           01EC  C3F201          JMP     @7
                           @6:
 160    1                            ELSE GOTO SOLAR;
                                                        ; STATEMENT # 160
           01EF  C3EE00          JMP     SOLAR
                           @7:
 161    1                  CALL TMP;
                                                        ; STATEMENT # 161
           01F2  CD7904          CALL    TMP
 162    1                        IF(TD<L)THEN GOTO BURNER;  /*IF TD<L & TS<I*/
                                                        ; STATEMENT # 162
           01F5  010000          LXI     B,L
           01F8  110B00          LXI     D,TD
           01FB  CD0000          CALL    @P0098
           01FE  D20402          JNC     @8
                                                        ; STATEMENT # 163
           0201  C34B01          JMP     BURNER
                           @8:
 164    1                            ELSE GOTO LOOP; /*IF TD>L & TS<L*/
                                                        ; STATEMENT # 164
           0204  C3B601          JMP     LOOP
                           @9:

 165    1              MIX:LPN=4;
                                                        ; STATEMENT # 165
                           MIX:

           0207  210400          LXI     H,4H
           020A  220500          SHLD    LPN
 166    1                  CALL CO('M');
                                                        ; STATEMENT # 166
           020D  0E4D            MVI     C,4DH
           020F  CD0000          CALL    CO
 167    1                  CALL CO('I');
                                                        ; STATEMENT # 167
           0212  0E49            MVI     C,49H
           0214  CD0000          CALL    CO
 168    1                  CALL CO('X');
                                                        ; STATEMENT # 168
           0217  0E58            MVI     C,58H
           0219  CD0000          CALL    CO
 169    1                  CALL CO(0AH);
                                                        ; STATEMENT # 169
           021C  0E0A            MVI     C,0AH
           021E  CD0000          CALL    CO
 170    1                  CALL CO(0DH);
                                                        ; STATEMENT # 170
           0221  0E0D            MVI     C,0DH
           0223  CD0000          CALL    CO
 171    1                  CALL TMP;
```

```
                                                       ; STATEMENT # 171
                      /*CHECK DAMPER POSITION; IF FULLY OPEN, THETA=50*/
          0226  CD7904         CALL    TMP
 172   1              IF((INPUT(0E6H) AND 1)=0)THEN THETA=0;
                                                       ; STATEMENT # 172
          0229  DBE6           IN      0E6H
          022B  E601           ANI     1H
          022D  FE00           CPI     0H
          022F  C23802         JNZ     @10
                                                       ; STATEMENT # 173
          0232  210000         LXI     H,0H
          0235  220100         SHLD    THETA
                          @10:
 174   1              IF((INPUT(0E6H) AND 2)=0)THEN THETA=50;
                                                       ; STATEMENT # 174
          0238  DBE6           IN      0E6H
          023A  E602           ANI     2H
          023C  FE00           CPI     0H
          023E  C24702         JNZ     @11
                                                       ; STATEMENT # 175

                      /*ITERATIVE PROCEDURE FOR DETERMINING
                      THEORETICAL SETTING ,GIVEN TS & TA*/
          0241  213200         LXI     H,32H
          0244  220100         SHLD    THETA
                          @11:

 176   1              N,A=32;
                                                       ; STATEMENT # 176
          0247  212000         LXI     H,20H
          024A  220300         SHLD    N
          024D  7D             MOV     A,L
          024E  321000         STA     A
 177   1              CALL CO('I');
                                                       ; STATEMENT # 177
          0251  0E49           MVI     C,49H
          0253  CD0000         CALL    CO
 178   1              CALL CO('T');
                                                       ; STATEMENT # 178
          0256  0E54           MVI     C,54H
          0258  CD0000         CALL    CO
 179   1              CALL CO('E');
                                                       ; STATEMENT # 179
          025B  0E45           MVI     C,45H
          025D  CD0000         CALL    CO
 180   1              CALL CO('R');
                                                       ; STATEMENT # 180
          0260  0E52           MVI     C,52H
          0262  CD0000         CALL    CO
 181   1              CALL CO(0DH);
                                                       ; STATEMENT # 181
          0265  0E0D           MVI     C,0DH
          0267  CD0000         CALL    CO
 182   1              CALL CO(0AH);
                                                       ; STATEMENT # 182
```

```
              026A  0E0A              MVI    C,0AH
              026C  CD0000            CALL   CO
 183    1            ITER:    DO I=0 TO 5;
                                                    ; STATEMENT # 183
                          ITER:
              026F  210F00            LXI    H,I
              0272  3600              MVI    M,0H
                          @36:
              0274  3E05              MVI    A,5H
              0276  210F00            LXI    H,I
              0279  BE                CMP    M
              027A  DA2D03            JC     @37
 184    2                 TD=(TS*10/KS(N)+TA*10/KA(N))*10;
                                                    ; STATEMENT # 184
              027D  210700            LXI    H,TS
              0280  CD0000            CALL   @P0031
              0283  E5                PUSH   H         ; 1
              0284  2A0300            LHLD   N
              0287  010400            LXI    B,KS
              028A  29                DAD    H
              028B  09                DAD    B
              028C  4E                MOV    C,M
              028D  23                INX    H
              028E  46                MOV    B,M
              028F  D1                POP    D         ; 1
              0290  CD0000            CALL   @P0030
              0293  210900            LXI    H,TA
              0296  D5                PUSH   D         ; 1
              0297  CD0000            CALL   @P0031
              029A  E5                PUSH   H         ; 2
              029B  2A0300            LHLD   N
              029E  016800            LXI    B,KA
              02A1  29                DAD    H
              02A2  09                DAD    B
              02A3  4E                MOV    C,M
              02A4  23                INX    H
              02A5  46                MOV    B,M
              02A6  D1                POP    D         ; 2
              02A7  CD0000            CALL   @P0030
              02AA  E1                POP    H         ; 1
              02AB  19                DAD    D
              02AC  CD0000            CALL   @P0033
              02AF  220B00            SHLD   TD
 185    2                 A=SHR(A,1);
                                                    ; STATEMENT # 185
              02B2  3A1000            LDA    A
              02B5  B7                ORA    A
              02B6  1F                RAR
              02B7  321000            STA    A
 186    2                 IF(TD>U)THEN N=N-SHL(A,1)+A;
                                                    ; STATEMENT # 186
              02BA  112200            LXI    D,U
              02BD  CD0000            CALL   @P0102
              02C0  D2DA02            JNC    @12
                                                    ; STATEMENT # 187
              02C3  3A1000            LDA    A
              02C6  87                ADD    A
```

```
          02C7  110300          LXI     D,N
          02CA  CD0000          CALL    @P0101
          02CD  E5              PUSH    H           ; 1
          02CE  2A1000          LHLD    A
          02D1  2600            MVI     H,0
          02D3  C1              POP     B           ; 1
          02D4  09              DAD     B
          02D5  EB              XCHG
          02D6  2B              DCX     H
          02D7  73              MOV     M,E
          02D8  23              INX     H
          02D9  72              MOV     M,D
                        @12:
188   2            IF(TD<L)THEN N=N+A;
                                                    ; STATEMENT # 188
          02DA  010000          LXI     B,L
          02DD  110B00          LXI     D,TD
          02E0  CD0000          CALL    @P0098
          02E3  D2F402          JNC     @13
                                                    ; STATEMENT # 189
          02E6  3A1000          LDA     A
          02E9  110300          LXI     D,N
          02EC  CD0000          CALL    @P0014
          02EF  EB              XCHG
          02F0  2B              DCX     H
          02F1  73              MOV     M,E
          02F2  23              INX     H
          02F3  72              MOV     M,D
                        @13:
190   2            CALL CO(' ');
                                                    ; STATEMENT # 190
          02F4  0E20            MVI     C,20H
          02F6  CD0000          CALL    CO
191   2            CALL CO('N');
                                                    ; STATEMENT # 191
          02F9  0E4E            MVI     C,4EH
          02FB  CD0000          CALL    CO
192   2            CALL CO('=');
                                                    ; STATEMENT # 192
          02FE  0E3D            MVI     C,3DH
          0300  CD0000          CALL    CO
193   2            CALL NMOUT(LOW(N));
                                                    ; STATEMENT # 193
          0303  2A0300          LHLD    N
          0306  7D              MOV     A,L
          0307  4F              MOV     C,A
          0308  CD0000          CALL    NMOUT
194   2            CALL CO(0DH);
                                                    ; STATEMENT # 194
          030B  0E0D            MVI     C,0DH
          030D  CD0000          CALL    CO
195   2            CALL CO(0AH);
                                                    ; STATEMENT # 195
          0310  0E0A            MVI     C,0AH
          0312  CD0000          CALL    CO
196   2            IF(N>49)THEN N=48;
                                                    ; STATEMENT # 196
```

```
          0315  3E31              MVI     A,31H
          0317  210300            LXI     H,N
          031A  CD0000            CALL    @P0103
          031D  D22603            JNC     @14
                                                    ; STATEMENT # 197
          0320  213000            LXI     H,30H
          0323  220300            SHLD    N
                            @14:
 198   2                 END;
                                                    ; STATEMENT # 198
          0326  210F00            LXI     H,I
          0329  34                INR     M
          032A  C27402            JNZ     @36
                            @37:
 199   1                               DO WHILE (N<THETA);
                                                    ; STATEMENT # 199
                            @38:
          032D  010100            LXI     B,THETA
          0330  110300            LXI     D,N
          0333  CD0000            CALL    @P0098
          0336  D24603            JNC     @39
 200   2                               CALL BKWD;
                                                    ; STATEMENT # 200
          0339  CD5B06            CALL     BKWD
 201   2                               THETA=THETA-1;
                                                    ; STATEMENT # 201
          033C  2A0100            LHLD     THETA
          033F  2B                DCX      H
          0340  220100            SHLD     THETA
 202   2                               END;
                                                    ; STATEMENT # 202
          0343  C32D03            JMP     @38
                            @39:
 203   1                               DO WHILE(N>THETA);
                                                    ; STATEMENT # 203
                            @40:
          0346  110100            LXI     D,THETA
          0349  010300            LXI     B,N
          034C  CD0000            CALL    @P0098
          034F  D25F03            JNC     @41
 204   2                               CALL FWD;
                                                    ; STATEMENT # 204
          0352  CDCD05            CALL     FWD
 205   2                               THETA=THETA+1;
                                                    ; STATEMENT # 205
          0355  2A0100            LHLD     THETA
          0358  23                INX      H
          0359  220100            SHLD     THETA
 206   2                               END;
                                                    ; STATEMENT # 206
          035C  C34603            JMP     @40
                            @41:

 207   1                           CALL CO('T');
                                                    ; STATEMENT # 207
          035F  0E54              MVI     C,54H
          0361  CD0000            CALL    CO
```

```
208    1                                CALL CO('H');
                                                    ; STATEMENT # 208
          0364   0E48              MVI       C,48H
          0366   CD0000            CALL      CO
209    1                                CALL CO('E');
                                                    ; STATEMENT # 209
          0369   0E45              MVI       C,45H
          036B   CD0000            CALL      CO
210    1                                CALL CO('T');
                                                    ; STATEMENT # 210
          036E   0E54              MVI       C,54H
          0370   CD0000            CALL      CO
211    1                                CALL CO('A');
                                                    ; STATEMENT # 211
          0373   0E41              MVI       C,41H
          0375   CD0000            CALL      CO
212    1                                CALL CO('=');
                                                    ; STATEMENT # 212
          0378   0E3D              MVI       C,3DH
          037A   CD0000            CALL      CO
213    1                                CALL NMOUT(LOW(THETA));
                                                    ; STATEMENT # 213
          037D   2A0100            LHLD      THETA
          0380   7D                MOV       A,L
          0381   4F                MOV       C,A
          0382   CD0000            CALL      NMOUT
214    1                                CALL CO(0DH);
                                                    ; STATEMENT # 214
          0385   0E0D              MVI       C,0DH
          0387   CD0000            CALL      CO
215    1                                CALL CO(0AH);
                                                    ; STATEMENT # 215
          038A   0E0A              MVI       C,0AH
          038C   CD0000            CALL      CO

            /*THIS ROUTINE WILL MAINTAIN THE CORRECT DAMPER SETTING
                   FOR THE  DESIRED DRYER TEMPERATURE*/

216    1             MAINT:  OE4=(OE4 AND 0F8H) OR 6;
                                                    ; STATEMENT # 216
                          MAINT:
          038F   3A0000            LDA       OE4
          0392   E6F8              ANI       0F8H
          0394   F606              ORI       6H
          0396   320000            STA       OE4
217    1                OUTPUT(0E4H)=OE4;             /*SELECT TD*/
                                                    ; STATEMENT # 217
          0399   D3E4              OUT       0E4H
218    1                OUTPUT(0EAH)=1;
                                                    ; STATEMENT # 218
          039B   3E01              MVI       A,1H
          039D   D3EA              OUT       0EAH
219    1                OUTPUT(0EAH)=0;
                                                    ; STATEMENT # 219
          039F   3E00              MVI       A,0H
          03A1   D3EA              OUT       0EAH
220    1                CALL CO('M');
```

```
                                                        ; STATEMENT # 220
          03A3   0E4D             MVI      C,4DH
          03A5   CD0000           CALL     CO
221    1                 CALL CO('A');
                                                        ; STATEMENT # 221
          03A8   0E41             MVI      C,41H
          03AA   CD0000           CALL     CO
222    1                 CALL CO('I');
                                                        ; STATEMENT # 222
          03AD   0E49             MVI      C,49H
          03AF   CD0000           CALL     CO
223    1                 CALL CO('N');
                                                        ; STATEMENT # 223
          03B2   0E4E             MVI      C,4EH
          03B4   CD0000           CALL     CO
224    1                 CALL CO('T');
                                                        ; STATEMENT # 224
          03B7   0E54             MVI      C,54H
          03B9   CD0000           CALL     CO
225    1                 CALL CO(0AH);
                                                        ; STATEMENT # 225
          03BC   0E0A             MVI      C,0AH
          03BE   CD0000           CALL     CO
226    1                 CALL CO(0DH);
                                                        ; STATEMENT # 226
          03C1   0E0D             MVI      C,0DH
          03C3   CD0000           CALL     CO
227    1                 CALL DELAY;     /*10 SEC DELAY*/
                                                        ; STATEMENT # 227

          03C6   CD1606           CALL     DELAY
228    1                 CALL TMP;
                                                        ; STATEMENT # 228

            /*THE FOLLOWING ROUTINES COMPENSATE FOR ANY POSSIBLE INACCURACIES
               IN THE THEORETICAL CALCULATION FOR THE DAMPER SETTING*/

          03C9   CD7904           CALL     TMP
229    1            DCR:     IF((TD>U) AND (THETA>0)) THEN DO;
                                                        ; STATEMENT # 229
                       DCR:
          03CC   110200           LXI      D,U
          03CF   010300           LXI      B,TD
          03D2   CD0000           CALL     @P0098
          03D5   9F               SBB      A
          03D6   F5               PUSH     PSW      ; 1
          03D7   3E00             MVI      A,0H
          03D9   210100           LXI      H,THETA
          03DC   CD0000           CALL     @P0103
          03DF   9F               SBB      A
          03E0   C1               POP      B        ; 1
          03E1   48               MOV      C,B
          03E2   A1               ANA      C
          03E3   1F               RAR
          03E4   D2F103           JNC      @15
231    2                               THETA=THETA-1;
                                                        ; STATEMENT # 231
```

```
                03E7  2A0100          LHLD    THETA
                03EA  2B              DCX     H
                03EB  220100          SHLD    THETA
 232    2                                         CALL FWD;
                                                                 ; STATEMENT # 232
                03EE  CDCD05          CALL    FWD
 233    2                                         END;
                                                                 ; STATEMENT # 233
                              @15:
 234    1              INCR:    IF((TD<L) AND (THETA<50)) THEN DO;
                                                                 ; STATEMENT # 234
                              INCR:
                03F1  010000          LXI     B,L
                03F4  110B00          LXI     D,TD
                03F7  CD0000          CALL    @P0098
                03FA  9F              SBB     A
                03FB  F5              PUSH    PSW          ; 1
                03FC  3E32            MVI     A,32H
                03FE  110100          LXI     D,THETA
                0401  CD0000          CALL    @P0101
                0404  9F              SBB     A
                0405  C1              POP     B            ; 1
                0406  48              MOV     C,B
                0407  A1              ANA     C
                0408  1F              RAR
                0409  D21604          JNC     @16
 236    2                                         CALL FWD;
                                                                 ; STATEMENT # 236
                040C  CDCD05          CALL    FWD
 237    2                                         THETA=THETA+1;
                                                                 ; STATEMENT # 237
                040F  2A0100          LHLD    THETA
                0412  23              INX     H
                0413  220100          SHLD    THETA
 238    2                                         END;
                                                                 ; STATEMENT # 238
                              @16:
 239    1                  CALL TMP;
                                                                 ; STATEMENT # 239

                    /*THIS ROUTINE WILL MAINTAIN DESIRED DRYER TEMP INDEFINITELY
                        UNLESS SOLAR TEMP DROPS BELOW LOWER LIMIT;
                        IN WHICH CASE, PROGRAM RETURNS TO SOLAR ROUTINE*/

                0416  CD7904          CALL    TMP
 240    1                  IF(TS<=L)THEN GOTO SOLAR;
                                                                 ; STATEMENT # 240
                0419  110000          LXI     D,L
                041C  010700          LXI     B,TS
                041F  CD0000          CALL    @P0098
                0422  DA2804          JC      @17
                                                                 ; STATEMENT # 241
                0425  C3EE00          JMP     SOLAR

                    /*IF THE DRYER COMPARTMENT BECOMES TOO HOT OR TO COOL,
                    AND THE DAMPER IS AT MIN OR MAX SETTING RESPECTIVELY,
                    AN ALARM LIGHT WILL BE ACTIVATED AND THE PROGRAM HALTED*/
```

```
                               @17:

242    1                    IF((TD>U) AND (THETA=Ø))THEN GO TO ALARM;
                                                        ; STATEMENT # 242
          Ø428   11Ø2ØØ             LXI      D,U
          Ø42B   Ø12BØØ             LXI      B,TD
          Ø42E   CDØØØØ             CALL     @PØØ98
          Ø431   9F                 SBB      A
          Ø432   F5                 PUSH     PSW      ; 1
          Ø433   3EØØ               MVI      A,ØH
          Ø435   11Ø1ØØ             LXI      D,THETA
          Ø438   CDØØØØ             CALL     @PØ1Ø1
          Ø43B   B5                 ORA      L
          Ø43C   D6Ø1               SUI      1
          Ø43E   9F                 SBB      A
          Ø43F   C1                 POP      B        ; 1
          Ø44Ø   48                 MOV      C,B
          Ø441   A1                 ANA      C
          Ø442   1F                 RAR
          Ø443   D249Ø4             JNC      @18
                                                        ; STATEMENT # 243
          Ø446   C36DØ4             JMP      ALARM
                               @18:
244    1                    IF((TD<L) AND (THETA=5Ø))THEN GOTO ALARM;
                                                        ; STATEMENT # 244
          Ø449   Ø1ØØØØ             LXI      B,L
          Ø44C   11ØBØØ             LXI      D,TD
          Ø44F   CDØØØØ             CALL     @PØØ98
          Ø452   9F                 SBB      A
          Ø453   F5                 PUSH     PSW      ; 1
          Ø454   3E32               MVI      A,32H
          Ø456   11Ø1ØØ             LXI      D,THETA
          Ø459   CDØØØØ             CALL     @PØ1Ø1
          Ø45C   B5                 ORA      L
          Ø45D   D6Ø1               SUI      1
          Ø45F   9F                 SBB      A
          Ø46Ø   C1                 POP      B        ; 1
          Ø461   48                 MOV      C,B
          Ø462   A1                 ANA      C
          Ø463   1F                 RAR
          Ø464   D26AØ4             JNC      @19
                                                        ; STATEMENT # 245
          Ø467   C36DØ4             JMP      ALARM
                               @19:

246    1                    GO TO MAINT;
                                                        ; STATEMENT # 246
          Ø46A   C38FØ3             JMP      MAINT

247    1               ALARM:  OE4=OE4 AND ØBFH;    /*SET ALARM BIT*/
                                                        ; STATEMENT # 247
                               ALARM:
          Ø46D   3AØØØØ             LDA      OE4
          Ø47Ø   E6BF               ANI      ØBFH
          Ø472   32ØØØØ             STA      OE4
248    1                    OUTPUT(ØE4H)=OE4;    /*RAISE ALARM LINE*/
                                                        ; STATEMENT # 248
```

```
          0475  D3E4              OUT     0E4H
 249   1       END USDA;
                                                          ; STATEMENT # 249
          0477  FB                EI
          0478  76                HLT


MODULE INFORMATION:

     CODE AREA SIZE     = 06A4H   1700D
     VARIABLE AREA SIZE = 0011H     17D
     MAXIMUM STACK SIZE = 0006H      6D
     312 LINES READ
     0 PROGRAM ERROR(S)

END OF PL/M-80 COMPILATION
```

APPENDIX C

REFERENCES

REFERENCES

1. SBC 80/20 and SBC 80/20-4 Single Board Computers Hardware Reference Manual. 1978. Intel Corporation.

2. Twelve Bit Binary A/D Converter for Microprocessor Interfaces. Intersil Data Book. Intersil Corporation.

3. Linear Data Book. 1976. National Semiconductor Corporation.

4. Slo-Syn Stepping Motors/Controls. 1979. Superior Electric Company, Bristol, CT.

5. Keey, R. B. 1972. Drying: Principles and Practice. Pergamon Press, Oxford, England. 358 p.

6. Duffie, J. A. and Beckman, W. A. 1974. Solar Energy Thermal Processes. John Wiley and Sons, Inc. New York, N.Y. 386 p.